

# Sistema para el control, posicionamiento y direccionamiento de vehículos

## *System for the control, positioning and address of vehicles*

Alejandro Mario Velázquez Carralero· José Alejandro Segura Roque

Universidad de las Ciencias Informáticas

[amvelazquez@uci.cu](mailto:amvelazquez@uci.cu)

### **Resumen**

El posicionamiento y direccionamiento a través de GPS ha cobrado vital importancia en el mundo actual. Sus aplicaciones en la gestión de emergencias y control de recursos han contribuido al mejoramiento de la eficiencia en empresas, así como la disponibilidad de servicios que, en última instancia, redundan en el aumento de la calidad de vida y de atención a las personas.

En el presente trabajo se expone la implementación de un sistema de software para el posicionamiento y direccionamiento de vehículos mediante la información brindada por GPS. El mismo se basa en el empleo de estándares internacionales para la compresión, almacenamiento y transmisión de imágenes satelitales, así como para la representación de rutas, puntos de interés y recorridos en el tiempo.

Para su desarrollo se han utilizado librerías gratuitas y de código abierto compatibles con los estándares elegidos, y el marco de trabajo (framework) .Net de Microsoft.

Se pretende crear una herramienta que permita a los conductores conocer su posición actual y lograr direccionamientos hacia puntos de su interés. Las entidades que decidieran usarlo, tendrían un instrumento de gestión y control, que les permitiría no solo posicionar y direccionar sus vehículos, sino tener un registro de los recorridos de los mismos en el tiempo.

**Palabras clave:** GPS, GIS, Localización.

### **Abstract**

*In this paper we present the implementation of a software system for positioning and routing of vehicles through information provided by GPS. We rely on the use of international standards for compression, storage and transmission of satellite imagery, as well as the representation of routes, points of interest and tours in time.*

*We aim to create a tool that would allow drivers to know their current position and achieve directions to points of interest. The entities that choose to use it, could have an instrument of management and control, allowing them to not only position and route their vehicles, but have a record of the routes at the same time.*

**Key words:** GPS, GIS, Location

### **Introducción**

En las entidades estatales de nuestro país existen varios problemas relacionados con la imposibilidad de la ubicación exacta de vehículos, así como del registro en el tiempo de su recorrido. En ocasiones, los conductores de ambulancias no saben dónde se encuentra una dirección determinada a la cual deben llegar con urgencia, además la central de ambulancias no es capaz de determinar con exactitud cuál es la que se encuentra más cerca del lugar donde se necesita. Este tipo de situaciones pueden provocar que al paciente se le deteriore aún más su estado de salud y en algunos casos llegue a perder la vida.

En una situación similar, se encuentran las patrullas de la PNR con las cuales se podría lograr una mejor gestión de emergencias si se conociera el lugar, el camino y la distancia exacta que los separan de un lugar indicado. En otro ámbito tenemos al turismo, en el cual sería igualmente aplicable un sistema de este tipo. De forma general, en ninguna de nuestras entidades estatales se puede

conocer el recorrido realizado por los conductores de sus vehículos en un período de tiempo, posibilitando la utilización de los recursos del estado en beneficio personal.

Por todo lo anteriormente mencionado se plantea que si se desarrolla un Sistema de Control, Posicionamiento y Direccionamiento de Vehículos, utilizando la información brindada por los GPS (Sistema de Posicionamiento Global), mejorará la gestión de los vehículos por parte de las entidades estatales de nuestro país, así como el servicio que brindan las mismas a la población.

## **Materiales y métodos**

Analítico - sintético:

Nos ha permitido analizar las teorías y los documentos referentes al objetivo de nuestra investigación, permitiéndonos de esta forma la extracción de los elementos más importantes relacionados con nuestro objeto de estudio. Además de permitirnos construir nuestra teoría y nuestro camino a seguir, a partir del análisis detallado de cada uno de los documentos previamente mencionados.

Análisis histórico-lógico:

Este método nos ha ayudado a comprender como han evolucionado en el mundo los Sistemas de Direccionamiento, permitiéndonos conocer la esencia del objeto de estudio, investigando las leyes generales y primordiales de su funcionamiento.

Modelación:

Muy útil nos ha resultado este método, pues para la realización de nuestro sistema hemos utilizado RUP como metodología de desarrollo de software, con el cual se hace muy necesaria la creación de varios modelos, ya que estos nos permiten una reproducción ampliada de la realidad, además de permitirnos descubrir y estudiar nuevas relaciones y cualidades de nuestro objeto de estudio.

Observación:

Éste es el único método empírico que hemos usado para la realización de nuestro sistema debido a que todavía no contamos con un cliente. Nos ha resultado muy útil, pues casi todo lo que hemos podido crear, lo hemos hecho a través del registro visual de lo que ocurre en la situación real que nos ocupa.

Materiales Empleados.

Para la confección de dicha solución se utilizaron las siguientes tecnologías:

- WPF (Windows Presentation Foundation), para la creación de interfaces de usuario amigables y atractivas para los usuarios.
- WCF (Windows Communication Foundation), para crear aplicaciones distribuidas. Este es un nuevo modelo de programación unificado propuesto por Microsoft para la creación de aplicaciones distribuidas.
- C#, lenguaje nativo de la plataforma . Net, utilizado para la codificación.
- Microsoft Caching Application Block, utilizada para el trabajo con la cache en toda la aplicación.
- Network Load Balancing (NLB), tecnología de balanceo de cargas en la red (NLB) que distribuye el tráfico entrante hacia el clúster de servidores.

Resultados y discusión

El sistema propuesto consta de 3 partes fundamentales: un cliente inteligente, la aplicación controladora y el servidor.

El cliente inteligente se encontrará ubicado en el vehículo, permitiéndole al conductor conocer su posición en tiempo real y además el direccionamiento hacia un punto geográfico determinado por él o asignado por otra persona en funciones de expedidor.

La aplicación controladora, permitirá visualizar la posición en tiempo real de aquellos vehículos que se encuentran en el “dominio“ del controlador, posibilitará además interactuar con cada uno de los conductores, asignarles destinos, así como visualizar sus posibles rutas y recorridos en un período de tiempo.

El servidor por su parte, almacenará y gestionará todos los datos necesarios para el buen funcionamiento del sistema, desde el repositorio de imágenes hasta los datos de cada uno de los vehículos en su dominio.

El algoritmo de enrutamiento se diseñó para que fuera configurable a las necesidades y gustos de los posibles clientes. Mientras que en su versión básica se vale de estimaciones heurísticas de los mejores recorridos basados en modelos matemáticos probados internacionalmente (Manhattan, Chebyshev, Geometría del Taxi, etc.), en una versión más avanzada, podría utilizar heurísticas basadas además en el estado de las carreteras, número de semáforos, velocidad promedio en el recorrido, información en tiempo real de embotellamientos, etc.

Estas heurísticas vienen a servir de soporte para la aplicación del algoritmo A\* (A Star), que es en fin de cuentas el que usamos para determinar las rutas óptimas. El problema de algunos algoritmos de búsqueda en grafos informados, es que se guían exclusivamente por la función heurística, la cual puede no indicar el camino de coste más bajo, o por el coste real de desplazarse de un nodo a otro, pudiéndose dar el caso de que sea necesario realizar un movimiento de coste mayor para alcanzar la solución. Es por ello bastante intuitivo el hecho de que un buen algoritmo de búsqueda informada debería tener en cuenta ambos factores, el valor heurístico de los nodos y el coste real del recorrido.

Así, el algoritmo A\* utiliza una función de evaluación  $f(n) = g(n) + h(n)$ , donde  $h(n)$  representa el valor heurístico del nodo a evaluar, y  $g(n)$ , el coste real del camino recorrido para llegar a dicho nodo. A\* mantiene dos estructuras de datos auxiliares, que podemos denominar abiertos, implementado como una cola de prioridad (ordenada por el valor  $f(n)$  de cada nodo), y cerrados, donde se guarda la información de los nodos que ya han sido visitados. En cada paso del algoritmo, se expande el nodo que esté primero en abiertos, y en caso de que no sea un nodo objetivo, calcula la  $f(n)$  de todos sus hijos, los inserta en abiertos, y pasa el nodo evaluado a cerrados. El algoritmo es una combinación entre búsquedas del tipo “primero en anchura” con “primero en profundidad”: mientras que  $h(n)$  tiende a “primero en profundidad”,  $g(n)$  tiende a “primero en anchura”. De este modo, se cambia de camino de búsqueda cada vez que existen nodos más prometedores.

Para alimentar este grafo se le deben introducir solamente los puntos que sean significativos para el direccionamiento, a saber, los inicios y fines de las calles y las intersecciones entre ellas. Todos los demás puntos interiores, que definen a las calles geodésicamente, no son significativos, pues, si se puede ir desde un inicio hasta un fin, quiere decir que también se puede transitar hasta cualquier punto interior a éstos. Las intersecciones se determinan por puntos que están repetidos en dos o más calles, sin embargo, la posibilidad de errores humanos a la hora de introducir los datos, provoca que existan puntos repetidos sin que estos sean una intersección. Todo esto implica un trabajo de discriminación para determinar las intersecciones verdaderas. Después de hecho todo esto es que se puede proceder a introducir todos estos puntos al grafo y establecer las relaciones entre ellos.

En el caso de las imágenes, el formato de compresión y almacenamiento elegido fue ECW (Enhanced Compressed Wavelet), que se ha ido convirtiendo poco a poco en el estándar de facto en este tipo de aplicaciones, a pesar de ser de los últimos formatos en aparecer.

ECW comprime transformando las imágenes al espacio wavelet usando Transformaciones Wavelet Discretas (DWT) multi-nivel. El siguiente paso es la cuantización, que reduce la cantidad de información de la imagen, seguido por la fase de codificación, que es la que realmente comprime las imágenes del espacio wavelet. La imagen comprimida ECW se procesa línea a línea directamente a partir de la imagen original. Por tanto, la técnica de compresión ECW puede comprimir imágenes de cualquier tamaño usando relativamente poca memoria RAM, siendo extremadamente útil para la compresión de imágenes muy grandes. El

ECW usa una técnica de algoritmo "pipeline" recursivo que no requiere el uso de almacenamiento en disco mientras se realiza las transformaciones.

ECW comprime transformando las imágenes al espacio wavelet usando Transformaciones Wavelet Discretas (DWT) multi-nivel. El siguiente paso es la cuantización, que reduce la cantidad de información de la imagen, seguido por la fase de codificación, que es la que realmente comprime las imágenes del espacio wavelet. La imagen comprimida ECW se procesa línea a línea directamente a partir de la imagen original. Por tanto, la técnica de compresión ECW puede comprimir imágenes de cualquier tamaño usando relativamente poca memoria RAM, siendo extremadamente útil para la compresión de imágenes muy grandes. El ECW usa una técnica de algoritmo "pipeline" recursivo que no requiere el uso de almacenamiento en disco mientras se realiza las transformaciones.

Por último, una aplicación de este tipo, tiene que lidiar con estructuras de datos que en el mejor de los casos cuentan con cientos de miles de instancias, datos de éstas y sus relaciones. Además, el usuario común espera tiempos de respuesta muy cortos, sin reparar quizás en los cálculos que implican una sencilla operación de direccionamiento. Todo esto conlleva a que el tema del rendimiento se convierta en algo crucial.

Por lo tanto, toda estrategia o algoritmo tenía que pasar primero por un test de rendimiento, y el desarrollo en muchas ocasiones tuvo que buscar vías alternativas en pos de mejores tiempos de ejecución. Para lograr este objetivo nos basamos esencialmente en técnicas de indexación múltiple y hashing, de modo que, no se hiciera ni una sola búsqueda secuencial sino que, el dato que fuera, estuviera al alcance de una sola operación.

Segundo, utilizamos lo que denominamos después “cargas inteligentes” de datos por niveles, de modo que, en cada nivel solo se analizarían aquellos datos que tenían sentido en el mismo, por ejemplo, un direccionamiento a nivel de país solo tendría (y cargaría por tanto) datos de carreteras interprovinciales o interestatales según fuera el caso, y, a medida que fuera aumentando el nivel de detalle se irían cargando datos más específicos hasta llegar a cargar y representar todas las calles en un sector determinado.

Tercero, una aplicación de este tipo tiene que tener una gran prioridad para el hilo principal encargado de mantener actualizada la interfaz de usuario y por tanto, de la renderización de todas las imágenes y datos. Sin embargo, todas las operaciones que involucren búsquedas de imágenes y/o datos al servidor, tienen que realizarse en un segundo plano y por tanto, en hilos de ejecución paralelos al principal. Ahora, para que la interfaz de usuario se actualice de manera más eficiente, estos datos deben tener la mayor granularidad posible, esto implica que la cantidad de hilos de ejecución aumente excesivamente.

Por tanto, se hizo necesaria la implementación de un “administrador” de hilos de ejecución que llamamos ThreadManager, que permitiría definir políticas referentes a la cantidad máxima de hilos concurrentes activos. Además mantendría en una cola con prioridad los hilos en espera, para que, en caso de la llegada de nuevos hilos de ejecución con más urgencia, estos pudieran ser ubicados en correspondencia con su nivel de prioridad. Como resultado, se logró mantener la interfaz de usuario con un nivel de respuesta adecuado y al mismo tiempo tener un control absoluto de la cantidad y prioridad de hilos a ejecutar.

Cuarto, teniendo en cuenta que la transmisión de imágenes se hace desde un servicio web, que la serialización a XML de datos binarios es extremadamente ineficiente, y que la implementación básica que traen los servicios web de .Net de los estándares de SOAP 1.1 no incluye ninguna de las especificaciones WS-\*, usamos en principio los WSE 3.0 (Web Services Enhancements) y la implementación de los mismos de MTOM (Message Transmission Optimization Mechanism); lográndose que la transmisión de imágenes desde el servicio web fuera mediante serialización binaria y por lo tanto más eficiente. Pero, al migrar nuestro servidor a WCF (Windows Communication Foundation) pudimos lograr esto de una manera más simple y declarativa.

Quinto, con el fin de reducir la transmisión innecesaria de imágenes establecimos tres niveles de caché, el primero en la memoria del dispositivo o computadora cliente donde estarían las imágenes usadas más recientemente, el segundo en el dispositivo de

almacenamiento del cliente, que podría ser tarjetas de memorias Flash (por ejemplo memorias SD), discos duros, etc. Y el tercero estaría en la memoria de los servidores.

Todas estas características permitirán que nuestro sistema pueda llegar a brindar prestaciones semejantes a la de los sistemas de este tipo a nivel mundial, además de proporcionar el recorrido realizado por un vehículo en un período de tiempo determinado, la cual constituye una funcionalidad no muy común en este tipo de software.

Finalmente se desarrolló un sistema informático con la capacidad de gestionar, posicionar y direccionar vehículos basado en la información de dispositivos GPS. Todos los datos que usa o genera se acogen a los estándares internacionales que rigen este tipo de sistemas, tanto para la compresión de imágenes, como para la representación de rutas y puntos de interés. Estos estándares han sido seleccionados después de un análisis exhaustivo de las tendencias y recomendaciones emitidas por las organizaciones internacionales dedicadas al tema, como el Open GIS Consortium.

Hemos propuesto una arquitectura basada en las posibilidades reales de nuestro país, y donde es posible incluso prescindir de algunas de las partes constituyentes en dependencia de presupuestos, tipos de dispositivos y conectividad. Las interfaces de usuario han sido diseñadas con un balance entre el atractivo visual y el aprovechamiento de la experiencia del usuario.

Los algoritmos de direccionamiento están abiertos a la extensión y personalización sobre la base de las necesidades específicas en un escenario de despliegue y uso determinados. Estableciendo, no obstante, una base algorítmica de direccionamiento, usando para ello los modelos de estimación heurística de recorridos más populares en el mundo en este tipo de sistemas.



Fig. 1. Aplicación Controladora.

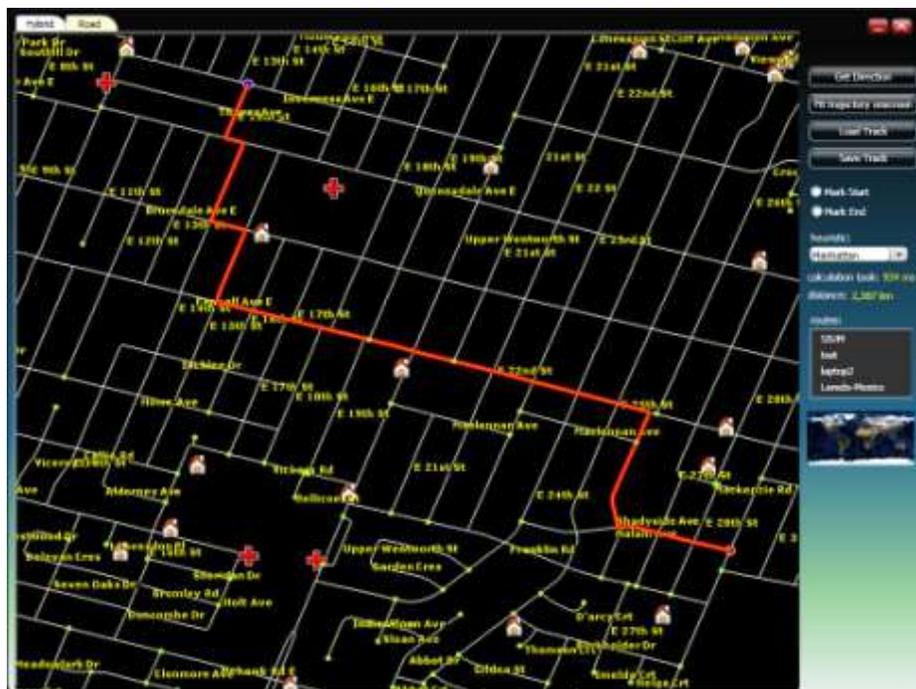


Fig. 2. Enrutamiento.

## Conclusiones

Con este trabajo se desarrolló un sistema informático con la capacidad de gestionar, posicionar y direccionar vehículos basado en la información de dispositivos GPS. Todos los datos que usa o genera se acogen a los estándares internacionales que rigen este tipo de sistemas, tanto para la compresión de imágenes, como para la representación de rutas y puntos de interés.

Estos estándares han sido seleccionados después de un análisis exhaustivo de las tendencias y recomendaciones emitidas por las organizaciones internacionales dedicadas al tema, como el Open GIS Consortium.

Hemos propuesto una arquitectura basada en las posibilidades reales de nuestro país, y donde es posible incluso prescindir de algunas de las partes constituyentes en dependencia de presupuestos, tipos de dispositivos y conectividad. Las interfaces de usuario han sido diseñadas con un balance entre el atractivo visual y el aprovechamiento de la experiencia del usuario.

Los algoritmos de direccionamiento están abiertos a la extensión y personalización sobre la base de las necesidades específicas en un escenario de despliegue y uso determinados. Estableciendo, no obstante, una base algorítmica de direccionamiento, usando para ello los modelos de estimación heurística de recorridos más populares en el mundo en este tipo de sistemas

## Referencias Bibliográficas

- ANDERSON, S. J. The Design and Implementation of Geographic Information Systems. John Wiley and Sons, 2003. 1100 p.
- ANDERSON, C. Essential Windows Presentation Foundation Addison-Wesley Professional 2007. 512 p. Microsoft .NET Development Series.
- ANDRADE, C. Professional WPF Programming: .NET Development with the Windows Presentation Foundation Wrox 2007. 480 p. Wrox Professional Guides.
- Algoritmo de búsqueda A\*, 2007  
<http://es.wikipedia.org/wiki/A%2A>
- AVENDAÑO, R. D. Acelerando Procesos de desarrollos, 2006  
<http://www.sistemasuni.edu.pe/AltaTecnologia/acelerandopro.html>

- BOSWORTH, A. XML, SOAP y datos binarios, 2003  
[http://www.microsoft.com/spanish/msdn/articulos/archivo/040403/voices/infoset\\_whitepaper.asp](http://www.microsoft.com/spanish/msdn/articulos/archivo/040403/voices/infoset_whitepaper.asp)
- CHAPPELL David. Windows Presentation Foundation (WPF), 2006  
<http://www.d2bnetwork.com/tecnologia/especializacion/wpf/default.aspx>
- DRAGICEVIC, S. Collaborative Geographic Information Systems. Idea Group Publishing, 2006. 1250 p.
- FOSTER, Dan .GPX: the GPS Exchange Format, 2004  
<http://www.topografix.com/gpx.asp>
- GRADY BOOCH, J. R., IVAR JACOBSON. The Unified Modeling Language User Guide. Second Edition. Addison Wesley 2005. p.
- Google Earth - la Tierra vista desde un satélite, 2007  
<http://google.dirson.com/o.a/google-earth/>
- HARVEY, F. A Primer of GIS: Fundamental Geographic and Cartographic Concepts The Guilford Press 2007. 300 p.
- KRUCHTEN, P. The Rational Unified Process: An Introduction. Third Edition. Addison Wesley, 2003. p.
- LARMAN, C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Third Edition. Addison Wesley, 2004. p.
- LOWY, J. Programming WCF Services O'Reilly Media, Inc., 2007. 634 p.
- MACDONALD, M. Pro WPF: Windows Presentation Foundation in .NET 3.0. Apress 2007. 1000 p. Pro.
- MOHAN R. C. Introducción al Servicio de Cluster Server de Microsoft (MSCS) en Windows .NET Server 2003, 2002
- MCMURTRY, C. Windows Communication Foundation Unleashed Sams 2007. 720 p. Unleashed
- ORTIZ, G. El funcionamiento del GPS: un repaso a los principales componentes, 2003  
<http://www.gabrielortiz.com/art.asp?Info=039>.
- PEIRIS, C. Pro WCF: Practical Microsoft SOA Implementation Apress 2007. 500 p. Pro.
- PENALVA, Javier P .Ford y Microsoft presentan Sync, 2007  
<http://xataka.com/2007/01/08-ces-2007-ford-y-microsoft-presentan-sync>
- ROJAS, N. R. Windows Communication Foundation (WCF), 2007  
<http://north18rojas.spaces.live.com/>
- SOAP Message Transmission Optimization Mechanism, 2005  
<http://www.w3.org/TR/soap12-mtom/>
- SHARMA, J. Frontiers of Geographic Information Technology. Springer-Verlag, 2006. 980 p.
- TRIGLAV, J. ECW: ¿Compresión Wavelet más allá del límite?, 2000  
[http://www.mappinginteractivo.com/plantilla-ante.asp?id\\_articulo=617](http://www.mappinginteractivo.com/plantilla-ante.asp?id_articulo=617)
- YEPES, E. L. Sistemas de Información geográfica, 2005  
<http://www.necochea.gov.ar/prensa128.htm>
- Volkswagen and Google develop revolutionary navigation system, 2007  
[http://www.automotoportal.com/article/Volkswagen\\_and\\_Google\\_develop\\_revolutionary\\_navigation\\_system](http://www.automotoportal.com/article/Volkswagen_and_Google_develop_revolutionary_navigation_system)  
<http://www.microsoft.com/spanish/msdn/articulos/archivo/100103/voices/wnsintroclustermcs.asp>