

DEPARTAMENTO DE INVESTIGACIÓN OPERATIVA
INSTITUTO DE COMPUTACIÓN
FACULTAD DE INGENIERÍA
UNIVERSIDAD DE LA REPÚBLICA ORIENTAL DEL URUGUAY

Sistemas de Información de Tráfico

Ing. Daniel Giosa Cuevas

Tutor y Supervisor: Msc. Ing. Omar Viera

Diciembre de 2003

Tesis para el título de Magíster en Informática
PEDECIBA

Resumen.

En este trabajo se estudia y se desarrolla un prototipo de un Sistema Avanzado de Información a Viajeros, que es un componente de los Sistemas de Transporte Inteligentes. Este componente es el encargado de diseminar la información que necesita una persona que debe realizar un viaje utilizando el transporte terrestre. El prototipo desarrollado utiliza un sitio web como forma de comunicar la información a los usuarios. Para el desarrollo fue necesario integrar distintos tipos de tecnologías como Sistemas de Información Geográfica, algoritmos de planificación de rutas, simulación de tráfico, bases de datos y páginas web.

El prototipo desarrollado demuestra la viabilidad técnica de estos sistemas. También se desarrollaron pruebas que demuestran que estos sistemas puede lograr ahorros de tiempo en los viajes y evitar congestiones.

Palabras claves: Sistemas de Transporte Inteligentes, Sistemas Avanzados de Información a Viajeros, Sistemas de Información Geográfica, Algoritmos de Planificación de rutas, Ruteo de Vehículos, Algoritmo A*, Simulación Microscópica del Tráfico de Vehículos, Servicios Web, Páginas Web.

Indice.

1	INTRODUCCIÓN.....	7
1.1	OBJETIVO DEL TRABAJO.....	7
1.2	TRABAJO REALIZADO.....	7
2	SISTEMAS DE TRANSPORTE INTELIGENTES.....	10
2.1	PROBLEMAS EN EL TRANSPORTE.....	11
2.2	ENFOQUES PARA RESOLVER LOS PROBLEMAS.....	11
2.3	¿QUÉ SON LOS SISTEMAS DE TRANSPORTE INTELIGENTES?.....	12
2.4	COMPONENTES.....	13
2.4.1	<i>Administración del Transporte.....</i>	<i>13</i>
2.4.2	<i>Sistemas Avanzados de Información a Viajeros.....</i>	<i>14</i>
2.4.3	<i>Manejo de Emergencias e Incidentes.....</i>	<i>14</i>
2.4.4	<i>Cobro Automático de Peajes.....</i>	<i>15</i>
2.4.5	<i>Transporte Público.....</i>	<i>15</i>
2.4.6	<i>Sistemas Rurales de Transporte Inteligente.....</i>	<i>16</i>
2.4.7	<i>Operación de Vehículos Comerciales.....</i>	<i>17</i>
2.4.8	<i>Vehículos Inteligentes.....</i>	<i>18</i>
2.5	ARQUITECTURA Y ESTÁNDARES.....	18
2.6	SISTEMAS DE TRANSPORTE INTELIGENTES EN EL MUNDO.....	19
2.6.1	<i>Japón.....</i>	<i>19</i>
2.6.2	<i>Francia.....</i>	<i>20</i>
2.6.3	<i>Estados Unidos.....</i>	<i>20</i>
3	SISTEMAS AVANZADOS DE INFORMACIÓN A VIAJEROS.....	21
3.1	DEFINICIÓN.....	21
3.1.1	<i>Medios para el acceso a la información.....</i>	<i>23</i>
3.1.2	<i>SAIV en Internet.....</i>	<i>25</i>
3.2	SAIV ESPECIALIZADOS EN INFORMACIÓN DE TRÁNSITO.....	27
3.2.1	<i>Qué servicios deben ofrecer.....</i>	<i>27</i>
3.2.2	<i>Qué servicios ofrecen.....</i>	<i>29</i>
3.3	SAIV ESPECIALIZADOS EN INFORMACIÓN DE TRÁFICO.....	30
3.3.1	<i>Por qué consultar información de tráfico.....</i>	<i>30</i>
3.3.2	<i>Qué servicios deben ofrecer.....</i>	<i>31</i>
3.3.3	<i>Qué servicios ofrecen.....</i>	<i>35</i>
3.3.4	<i>Usuarios de los SAIV.....</i>	<i>36</i>
3.4	INFORMACIÓN Y DATOS.....	37
3.5	ESTADO DEL ARTE EN SISTEMAS AVANZADOS DE INFORMACIÓN A VIAJEROS.....	38
3.5.1	<i>Tecnologías utilizadas actualmente.....</i>	<i>40</i>
4	UN PROTOTIPO DE SAIV DE TRÁFICO.....	42
4.1	INTRODUCCIÓN.....	42
4.2	ARQUITECTURA GENÉRICA DE UN SAIV.....	42
4.3	ARQUITECTURA DEL PROTOTIPO.....	44
4.4	BASE DE DATOS.....	44
4.4.1	<i>PostgreSQL.....</i>	<i>45</i>
4.4.2	<i>PostGIS.....</i>	<i>46</i>

4.4.3	<i>Organización de la Base de Datos</i>	46
4.4.4	<i>Consistencia de la Base de Datos</i>	48
4.5	SERVIDOR DE MAPAS DIGITALES.....	49
4.5.1	<i>Información geográfica digital</i>	49
4.5.2	<i>Representación de los elementos geográficos</i>	50
4.5.3	<i>Información geográfica sobre las calles</i>	50
4.5.4	<i>Servidor de Mapas</i>	52
4.5.5	<i>Interacción con el mapa</i>	54
4.5.6	<i>Configuración del Servidor de Mapas</i>	55
4.6	PLANIFICACIÓN DE LA RUTA.....	59
4.6.1	<i>Modelado del problema</i>	59
4.6.2	<i>Algoritmo de Dijkstra</i>	61
4.6.3	<i>Algoritmo A*</i>	61
4.6.4	<i>Implementación</i>	66
4.7	SITIO WEB.....	70
4.7.1	<i>Viajero Informado</i>	70
4.7.2	<i>Implementación</i>	73
5	SIMULACIÓN DEL TRÁFICO DE LA CIUDAD	80
5.1	SIMULACIÓN DEL TRÁFICO DE VEHÍCULOS.....	80
5.2	MODELO DE AUTOS SIGUIÉNDOSE.....	80
5.2.1	<i>Autómatas celulares</i>	81
5.3	MODELO PARA UNA SOLA CALLE.....	81
5.4	EXTENSIÓN DEL MODELO PARA UNA CIUDAD.....	84
5.4.1	<i>Calles</i>	84
5.4.2	<i>Intersecciones</i>	84
5.4.3	<i>Escala del modelo</i>	86
5.5	MEDIDAS.....	88
5.5.1	<i>Densidad</i>	89
5.5.2	<i>Flujo</i>	89
5.5.3	<i>Velocidad</i>	90
5.5.4	<i>Tiempo</i>	90
5.5.5	<i>Censores</i>	91
5.6	VEHÍCULOS.....	91
5.6.1	<i>Controladores</i>	91
5.6.2	<i>Rutas de los vehículos</i>	92
5.7	DETERMINACIÓN DEL ESTADO ESTABLE.....	93
5.7.1	<i>Tratamiento de callejones</i>	94
5.8	IMPLEMENTACIÓN.....	95
5.8.1	<i>Base de Datos</i>	95
5.8.2	<i>Motor del simulador</i>	97
5.8.3	<i>Visualización</i>	97
5.8.4	<i>Implementación</i>	102
5.9	HERRAMIENTAS DE SIMULACIÓN EVALUADAS.....	104
6	EXPERIMENTOS CON EL SISTEMA	106
6.1	PRUEBAS DE DESEMPEÑO.....	106
6.1.1	<i>Pruebas de desempeño</i>	106
6.1.2	<i>Arquitectura sugerida</i>	114
6.1.3	<i>Mejoras al desempeño</i>	115

6.2	EXPERIMENTOS REALIZADOS CON LOS ALGORITMOS DE PLANIFICACIÓN DE RUTAS.	117
7	CONCLUSIONES Y TRABAJO FUTURO.	119
7.1	APORTE DE ESTE TRABAJO.	119
7.2	CONCLUSIONES.	120
7.2.1	<i>STI y SAIV.</i>	120
7.2.2	<i>Viajero Informado.</i>	121
7.2.3	<i>Algoritmos de planificación de rutas.</i>	121
7.2.4	<i>Simulación de tráfico.</i>	122
7.3	TRABAJO FUTURO.	122
7.3.1	<i>Algoritmos.</i>	122
7.3.2	<i>Viajero Informado.</i>	123
7.3.3	<i>Simulación.</i>	124
	BIBLIOGRAFÍA.	126
	APÉNDICE A - DESCRIPCIÓN DE LAS TABLAS DE LA BASE DE DATOS.	133
	APÉNDICE B - MANEJO DE PARÁMETROS.	140
	APÉNDICE C - RESULTADOS DE LOS EXPERIMENTOS DE PLANIFICACIÓN DE RUTAS.	147

Agradecimientos.

Culminar este proyecto fue un largo proceso, que seguramente no hubiera podido realizar sin el apoyo de las personas que me rodean. Quisiera agradecer muy especialmente al tutor y supervisor de este trabajo Msc. Omar Viera que siempre me animó a seguir adelante. También quiero agradecer: a mi familia por apoyarme incondicionalmente en todos mis proyectos. A Gimena, por todo el cariño y paciencia que ha tenido conmigo. Al apoyo de mis amigos y compañeros de trabajo en Artech, que ha sido fundamental para poder seguir adelante. A Msc. Libertad Tansini e Ing. Gonzalo Tejera por realizar este viaje juntos. Al Dr. Héctor Cancela por sus valiosos comentarios sobre este informe. Al Departamento de Investigación Operativa, al Instituto de Computación y a la Facultad de Ingeniería, por continuar formándome y al PEDECIBA, por darme esta oportunidad.

Quiero agradecer a todos los que hicieron posible que haya podido contar con las herramientas software y los datos que necesité para este proyecto. Ellos son: ARTech por las licencias de Genexus. Leonardo Loureiro de Ingenieros Consultores Asociados por ofrecerme mapas y herramientas de software. La Intendencia Municipal de Montevideo que publicó los mapas digitales que se utilizan en este trabajo. A las comunidades de desarrolladores de software libre que se utilizan en este proyecto.

Dedico este trabajo a mi abuela Gladys y a la memoria de mis abuelos Luis y Libia.

Daniel Giosa Cuevas

1 Introducción.

Los problemas de congestión, seguridad y eficiencia que existen en los sistemas de transporte de algunos países causan enormes pérdidas económicas, daño al medio ambiente y numerosos accidentes de tráfico. A partir de la década del 80, la disciplina del transporte comprendió que las soluciones tradicionales que se estaban aplicando no serían capaces de resolver esta problemática. Estas soluciones típicamente implicaban la construcción de nueva infraestructura como construir o ampliar caminos y calles. Entonces surge la idea de aplicar los últimos avances de la tecnología para dar soluciones más efectivas al transporte. A partir de esa idea, la disciplina del transporte se renueva creando los Sistemas de Transporte Inteligentes, que es un conjunto de sistemas que resuelven los problemas asociados al transporte terrestre. Durante la década del 80 y 90 se produce una etapa de investigación y desarrollo con algunas implantaciones experimentales. Hacia finales de los 90 y principios del 2000, la disciplina se encuentra en pleno auge, lográndose exitosas implantaciones de componentes en distintas áreas. Sin embargo, aún queda mucho trabajo por hacer para lograr los niveles de implantación que quisieran tener los países más involucrados en el tema.

1.1 Objetivo del trabajo.

Los Sistemas de Transporte Inteligentes (STI) están divididos en distintos componentes que se encargan de resolver problemas específicos del transporte. En este trabajo se estudia uno de los componentes conocido como Sistemas Avanzados de Información a Viajeros (SAIV). Este componente es el encargado de difundir la información del tráfico y tránsito a todas las personas que deban realizar un viaje utilizando el sistema de transporte terrestre. Para llevar la información a los usuarios, se pueden utilizar distintos medios, por ejemplo teléfonos y televisión, entre otros. En este trabajo interesan particularmente los sistemas que utilizan Internet y sitios web como modo de diseminar la información. A principios del año 2000, que fue cuando comenzó este trabajo, los sitios web eran un medio poco tradicional de diseminar este tipo de información y en las implementaciones desarrolladas existían notables carencias en la información y servicios que se ofrecían, como planificación de rutas y mapas interactivos, entre otras. Esta situación alentó a definir como objetivo del trabajo: *desarrollar un prototipo de un Sistema Avanzado de Información a Viajeros que utilice un sitio web para ofrecer información y servicios de tráfico.*

El desarrollo del prototipo pretende servir como un guía inicial para que cuando estos sistemas comiencen a desarrollarse o a mejorarse, ya exista un trabajo que cubra los aspectos esenciales que deben tenerse en cuenta en el sistema.

1.2 Trabajo realizado.

Este trabajo comenzó a gestarse a principio del año 2000. La idea inicial del trabajo era investigar sobre algoritmos de ruteo de vehículos que no solo utilizaran

información geográfica para computar las rutas, sino que pudieran tener en cuenta otro tipo de información por ejemplo el nivel de tráfico existente. Inicialmente se estudia el tema de Guiar por la Ruta (Route Guidance), que consiste en planificar un recorrido desde un origen a un destino en una ciudad y guiar al conductor del vehículo por el recorrido, indicándole cuando debe de doblar y en que dirección. A medida que se profundiza en el estudio del tema, cada vez se hace más claro que el área de Guiar por la Ruta es parte del contexto de los Sistemas de Transporte Inteligentes. Entonces se decide investigar más en profundidad ese tema y ver en que consiste. Los Sistemas de Transporte Inteligentes se estudian en el Capítulo 2 de este informe. Luego de que esta etapa inicial de investigación finaliza, se realiza un informe que indica los conocimientos adquiridos hasta el momento y da pautas sobre las líneas de investigación que se deben seguir.

Luego de culminar esta primer etapa de aproximación al tema, la investigación continúa en el área de sistemas de navegación en vehículos, que son dispositivos incorporados en el vehículo que asisten en la navegación. A partir del estudio de este tema, cambia la idea de desarrollar un sistema de Guiar por la Ruta y se decide investigar uno de los componentes de los Sistemas de Transporte Inteligentes, los Sistemas Avanzados de Información a Viajeros, que una de las funciones que cumplen es Guiar por la Ruta. A principios del año 2000 distintos informes indican que las implementaciones de los Sistemas Avanzados de Información a Viajeros existentes, que utilizan sitios web, tenían carencias importantes, por ejemplo: [1], [2], [3] y [4]. A partir de esta situación se decide desarrollar un Sistema Avanzado de Información a Viajeros que utilice un sitio web como medio para diseminar la información. Los Sistemas Avanzados de Información a Viajeros se estudian en el Capítulo 3 de este informe.

Durante el año 2001 se continúa estudiando los Sistemas de Transporte Inteligentes y los Sistema Avanzado de Información a Viajeros y comienza el estudio específico de los componentes que serán necesario desarrollar. Ese año se desarrolla una primera versión de los algoritmos de planificación de rutas. Sobre finales del 2001 se realiza una presentación del avance del trabajo en las VII Jornadas de Informática e Investigación Operativa, en el Instituto de Computación de la Facultad de Ingeniería, UDELAR.

Para el año 2002 ya se tiene diseñada la arquitectura general del prototipo y comienza una etapa principalmente de desarrollo. Se desarrolla un sitio web llamado Viajero Informado que permite obtener información de tráfico, ver mapas y realizar planificaciones de rutas. El sitio web es integrado con un servidor de mapas web, con los algoritmos de planificación de rutas y con bases de datos que soportan el manejo de información geográfica. Los detalles de la implementación del prototipo del Sistema Avanzado de Información a Viajeros se estudian en detalle en el Capítulo 4.

En el año 2003 se desarrolla un simulador que genera información de tráfico, para que los algoritmos de planificación de rutas puedan utilizarla en las planificaciones. Luego de una etapa de investigación en el tema de simulación de tráfico se decide desarrollar un simulador microscópico utilizando Automatas Celulares. Inicialmente se desarrolla la parte central del simulador que permite mover los vehículos por las calles y luego

se agrega una interfaz gráfica y el almacenamiento de datos en la Base de Datos. Este tema es abordado en el Capítulo 5 de este informe.

Finalmente, se realizan algunas pruebas con las aplicaciones desarrolladas, que se detallan en el Capítulo 6.

2 Sistemas de Transporte Inteligentes.

El transporte terrestre es un factor muy importante en la vida de un país. Es necesario para sustentar el crecimiento económico y satisfacer las necesidades de movilidad de la comunidad. Los países hacen un esfuerzo económico muy importante para lograr un sistema de transporte que permita el desarrollo económico y social. Sin embargo el transporte enfrenta, generalmente, serios problemas que se repiten en forma similar todos los países del mundo. Problemas de congestión, seguridad, contaminación del medio ambiente y el alto costo en el mantenimiento de la infraestructura, son solo un ejemplo.

Uruguay no es ajeno a esta realidad mundial. Los accidentes de tránsito están considerados como la principal causa de muerte, principalmente entre personas jóvenes. Se producen aproximadamente 400 muertes por año y la cifra no tiende a disminuir [5]. En materia de congestión también hay problemas, las dificultades en la circulación en la zona céntrica de Montevideo, la capital del país, han llevado a las autoridades a implantar tarifas de estacionamiento como forma de desalentar el uso de vehículos particulares. En cuanto a la construcción y mantenimiento de las rutas y puentes del país, que generalmente se llevaban a cabo utilizando financiación pública, últimamente el estado recurre a la financiación privada, por el alto costo que esto le implica al estado hacerlo.

Para solucionar estos problemas, surge a partir de los años 80 en los países económicamente más desarrollados como Estados Unidos y los países europeos, una nueva disciplina en el transporte llamada Sistemas de Transporte Inteligentes, que busca mejorar la eficiencia y la seguridad del sistema. Aplicando avances tecnológicos en distintas áreas como electrónica, sistemas de información y satélites los Sistemas de Transporte Inteligentes, prometen revolucionar todos los aspectos del transporte.

En este Capítulo se estudian los problemas más comunes en el transporte y como los Sistemas de Transporte Inteligentes intentan solucionarlos.

2.1 Problemas en el transporte.

Dentro del transporte hay distintos tipos de problemas, pero dos de ellos son los más comunes y apremiantes, la congestión y la seguridad.

La congestión se refiere a cuando existen demasiados vehículos circulando por las calles y como consecuencia la velocidad de circulación tiende a disminuir, incluso llegando a detener completamente al tráfico. Este fenómeno se da debido a que la capacidad de las calles se ve excedido y tiene consecuencias negativas en distintos aspectos. Por un lado se desperdicia combustible y se contamina el aire. Las agendas de las personas y los horarios del transporte público se ven alterados. Además las personas desperdician incontables horas productivas, causando perjuicios económicos a ellos mismos y al país.

Otro problema muy grave tiene que ver con la seguridad. Cada año se producen innumerables accidentes de tránsito que causan graves daños a las personas, a la sociedad y a la economía. Estas situaciones no son exclusivas de Uruguay, también se dan en distinta medida en distintos países del mundo como Estados Unidos, Francia, Alemania, Japón y Corea, entre otros [6].

2.2 Enfoques para resolver los problemas.

El enfoque tradicional que se ha aplicado sistemáticamente para resolver los problemas de movilidad ha sido el de construir o ampliar las calles y caminos existentes. Si bien este enfoque ha resultado exitoso en épocas pasadas, hoy en día no resulta tan efectivo como lo fue antes. Esto se debe principalmente, al costo que tienen este tipo de obras, a las trabas impuestas para el cuidado del medio ambiente y el rechazo de la población a este tipo de proyectos [6]. Por otro lado en las ciudades, que es donde se encuentran con mayor frecuencia los problemas de congestión, ampliar la infraestructura no es una buena solución ya que resulta muy costoso o eventualmente imposible y el beneficio obtenido es escaso. Esto no quiere decir que en algunos casos ampliar la infraestructura no sea la solución correcta, simplemente no es una regla general.

En materia de seguridad las soluciones comúnmente intentan la prevención de accidentes. Generalmente se explicita la ordenanza del tráfico, estableciendo señalización o semáforos. Otra tarea importante de prevención es el control del cumplimiento de las ordenanzas de tránsito, principalmente en cuanto al exceso de velocidad y respeto de la señalización existente.

Para los expertos, la aplicación de la tecnología en el transporte les ofrece la posibilidad de revolucionar la disciplina. La tecnología ya ha sido utilizada en el tráfico por muchas décadas, principalmente enfocada a resolver problemas específicos de seguridad y ordenamiento. Hasta los años 80 la tecnología utilizada consistía principalmente en semáforos, inventados en los años 20 para ordenar el tráfico en las ciudades. En los años 60 surge el primer sistema de semáforos controlados por computadoras y los mensajes variables, que son paneles que despliegan mensajes relacionados con las condiciones del tráfico. A partir de la efectividad de estos

avances, se evidencia la conveniencia de aplicar tecnologías en el transporte. Es entonces cuando se comienza a hablar de Sistemas de Transporte Inteligentes. Durante los años 70 y 80, se estudia principalmente la viabilidad de posibles soluciones a distintas áreas y se desarrollan prototipos. Sobre finales de los 80, es cuando se consolida firmemente la disciplina, permitiendo que durante los años 90 se implanten con éxito distintos sistemas en varias partes del mundo.

2.3 ¿Qué son los Sistemas de Transporte Inteligentes?

Para lograr comprender qué son los STI se verán algunas definiciones dadas por distintas autoridades en el tema.

En la página web del Departamento de Transporte de Estados Unidos, en la sección de Preguntas Frecuentes, se da la siguiente respuesta a la pregunta ¿qué son los STI? [7]:

“Los Sistemas de Transporte Inteligentes representan el siguiente paso en la evolución de todo el sistema de transporte de una nación. Mientras que las tecnologías de la información y los avances en la electrónica continúan revolucionando todos los aspectos de la vida moderna, también son aplicados a la red de transporte. Esas tecnologías incluyen lo último en computadoras, electrónica, comunicaciones y sistemas de seguridad.”

Es decir que los sistemas de transporte evolucionarán hacia los STI, de la mano de los avances tecnológicos y transformarán la forma de vida de todas las personas. La siguiente definición, corresponde al Departamento de Transporte de Estados Unidos y aclarará el modo en que va lograrse la evolución [6]:

“Los Sistemas de Transporte Inteligentes recolectan, almacenan, procesan y distribuyen información relacionada con el movimiento de personas y bienes. Ejemplos incluyen sistemas para controlar el tráfico y el transporte público, manejar emergencias, brindar información a viajeros, realizar un control avanzado de vehículos y de seguridad, manejar operaciones con vehículos comerciales, cobrar automáticamente peajes y controlar en forma segura el cruce de barreras de tren.”

Esta definición deja en claro cual es el modo en que los STI podrán revolucionar los sistemas de transporte actuales, utilizando información. Teniendo en todo momento información confiable y actualizada de lo que esta ocurriendo en el sistema, se podrá actuar consecuentemente. La última definición la aporta la Intelligent Transportation Society of America [8]:

“La aplicación de tecnologías avanzadas en sensores, computadoras, electrónica, comunicaciones y estrategias de control, de manera integrada, para incrementar la seguridad y la eficiencia en el sistema de transporte terrestre.”

O sea que la integración de distintas tecnologías será la respuesta para resolver los problemas en el sistema de transporte terrestre.

Para resumir en una sola idea este concepto, se puede decir que, los Sistemas de Transporte Inteligentes son la evolución a los sistemas de transporte actuales, los cuales integrando distintas tecnologías de última generación e información, prometen mejorar el desempeño general del sistema, la calidad de vida de las personas y la economía.

Para poder brindar soluciones que acaparen todo el sistema de transporte, los STI dividen el sistema en distintas áreas. Cada una de esas áreas tiene asignado un componente encargado de estudiar y resolver los problemas específicos. En la siguiente Sección se presentan los principales componentes y las problemáticas que resuelven. Si se desea profundizar sobre estos componentes y otros que no se mencionan, se sugiere consultar [6].

2.4 Componentes.

Si bien los componentes que integran los STI ya están definidos, es muy probable que se incorporen otros nuevos a medida que la disciplina va evolucionando. Los componentes más importantes se describen a continuación y son: Administración del Transporte, Sistemas Avanzados de Información a Viajeros, Manejo de Emergencias, Cobro automático de peajes, Transporte Público, Sistemas Rurales de Transporte Inteligente, Operación de Vehículos Comerciales y Vehículos Inteligentes [6].

2.4.1 Administración del Transporte.

El área de Administración de Transporte sirve de ‘ojos’ del sistema, ya que se encarga de recolectar información del estado del sistema de transporte. Si bien utiliza esa información para emprender acciones sobre el sistema, también la deja disponible para que la utilicen otros componentes del STI. Por lo tanto el área de Administración del Transporte es un pilar fundamental en los STI. Los objetivos principales que tiene el área son: mejorar la seguridad, utilizar el sistema eficientemente y hacerlo predecible. Para lograr estos objetivos utiliza la infraestructura disponible, por ejemplo utilizando sistemas de semáforos adaptativos, sensores que indican el nivel de congestión del tráfico o brindando información a los usuarios. Si bien los sistemas de Administración de Transporte dependen del recurso que administran, por ejemplo, un túnel, una carretera o el sistema de semáforos, todos comparten las siguientes funciones básicas:

- Estimación del sistema. Consiste en obtener información sobre las condiciones actuales del sistema.
- Determinar una estrategia. Luego de que el estado del sistema está determinado, se necesita establecer una estrategia para cumplir los objetivos definidos. Esta es una tarea muy difícil ya que las condiciones en el sistema de transporte varían muy rápido y no es posible controlar a cada vehículo. Tradicionalmente esta tarea es realizada de un modo manual, aunque cada vez hay más herramientas disponibles.
- Ejecución de la estrategia. Luego que la estrategia ha sido determinada, se debe llevar a cabo. Para eso se utilizan los recursos de la infraestructura

disponibles, que pueden ser, adaptar el tiempo del cambio de luces de semáforos, enviar mensajes en carteles electrónicos o por estaciones de radio dedicadas a transmitir información de tráfico.

- Evaluación de la estrategia. Esta tarea completa el ciclo de las actividades del sistemas tal como se puede ver en la Figura 2-1. Esta función es importante debido a que en los sistemas de transporte las condiciones nunca son determinísticas y es necesario evaluar continuamente la efectividad de las estrategias. De este modo se pueden modificar las estrategias y comprender mejor cual es el mejor uso que se le puede dar a las herramientas disponibles.

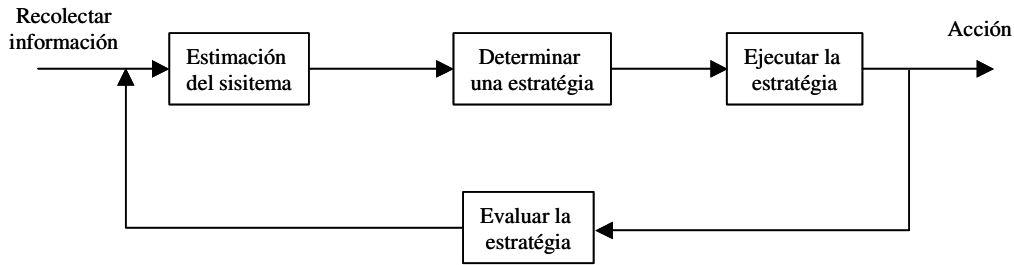


Figura 2-1. Funciones básicas de la administración del transporte.

2.4.2 Sistemas Avanzados de Información a Viajeros.

Esta área se encarga de diseminar al público la información recolectada por el área de Administración del Transporte. Disponiendo de esa información los usuarios puede hacer un uso más eficiente y seguro del sistema de transporte. Los Sistemas Avanzados de Información a Viajeros se distinguen dependiendo del tipo de información que brindan. Un tipo de sistema brindan información de tráfico y está orientado a los conductores de vehículos. El otro tipo de sistema existente brinda información sobre el tránsito y está orientado a las personas que se movilizan utilizando los distintos medios de transporte público. El desafío principal en esta área es poder llevar la información a las personas en el momento y lugar que la necesitan. Por ejemplo la información puede necesitarse antes de comenzar un viaje o en el medio de una congestión. Por otra lado está el medio en el que se accede a la información. Eso depende de varios factores, por ejemplo, si la persona está conduciendo su vehículo, podría acceder a la información mediante un teléfono celular, radio, dispositivos en el vehículo o por computadoras portátiles de bolsillo con conexión inalámbrica. En este trabajo se estudia esta última opción, es decir el acceso a información y herramientas de tránsito que son publicadas en sitios web y eventualmente pueden ser accedidas mediante computadoras portátiles desde la ruta con conexión inalámbrica. En el siguiente Capítulo se profundiza más en el estudio de los Sistemas Avanzados de Información a Viajeros.

2.4.3 Manejo de Emergencias e Incidentes.

Los sistemas de Manejo de Emergencias e Incidentes tienen como objetivo detectar incidentes y enviar la asistencia necesaria, lo más pronto posible. Los incidentes son

definidos como eventos que distorsionan el flujo normal del tráfico. Los incidentes incluyen, accidentes, obras de construcción o condiciones climáticas adversas.

La detección de incidentes tiene dos formas de operar: mediante dispositivos de seguridad personal o mediante la detección automática de colisiones. Los dispositivos de seguridad personal son utilizados por el usuario para enviar alertas al Centro de Administración de Tráfico indicando algún incidente, por ejemplo un desperfecto mecánico en su vehículo.

Por otro lado la detección automática de colisiones tiene como objetivo detectar choques de vehículos, en forma rápida y certera. Para eso se instalan detectores en los puntos más propicios a accidentes como carreteras de alta velocidad y cruces de calles. Las tecnologías más utilizadas son cámaras de video, detectores por ciclos de inducción (en Inglés llamados Inductive Loop Detectors) o micrófonos [9]. Cuando se informa de un accidente al Centro de Administración del Transporte, éste reacciona tratando de que el servicio requerido llegue al lugar lo más pronto posible. Para eso considera la flota de vehículos disponible, determinando cual es punto de salida más adecuado. Se planifica la ruta del vehículo teniendo en cuenta los niveles de congestión del momento y los semáforos ajustan el cambio de luces para que el vehículo pueda pasar sin detenerse [6].

2.4.4 Cobro Automático de Peajes.

En esta área se encarga de automatizar el cobro del peajes. Los beneficios de que el cobro de peajes esté automatizado son: reducir las colas, aumentar la tasa de vehículos que pasan, reducir las emisiones de gases, reducir el costo del sistema de peajes y facilitar el servicio al usuario, entre otras [9]. Para lograrlo existen distintas tecnologías, por ejemplo la Identificación Automática de Vehículos utiliza dispositivos electrónicos que identifican a los vehículos y registran automáticamente el pasaje del vehículo por el peaje. También se utilizan cámaras que registran la patente del vehículo, en el caso de que no tenga instalada el dispositivo de identificación [9]. Los sistemas de Cobro Automático de Peajes, son muy importantes en lugares donde existen muchos peajes o están ubicados en lugares de alto tránsito.

2.4.5 Transporte Público.

Los sistemas encargados del Transporte Público tienen como objetivo resolver distintos tipos de problemas que presenta el transporte actual, principalmente de ómnibus. Por ejemplo [9]:

- Problemas operativos como la variabilidad en los tiempos de las rutas o conocer si un ómnibus esta atrasado o cumple con el horario establecido.
- Problemas de seguridad, no se conoce donde está localizado un ómnibus si tiene un accidente
- Problemas en el mantenimiento de la flota, como desperfectos mecánicos.
- Problemas en el servicio dado al usuario, los usuarios perciben que los ómnibus no están coordinados y son lentos.
- Problemas de planificación, no existen datos, es muy costoso obtenerlos, no son confiables y no pueden utilizarse en la planificación.

Para revertir esta situación los sistemas de Transporte Público apuntan a la recolección de información para poder tener un mejor control del sistema. Para eso se utilizan distintas tecnologías que, por un lado obtienen la información y por otro la procesaran llevándola a quien la requiera, ya sean las compañías de ómnibus, los usuarios o los Sistemas de Manejo de Emergencias. Para la localización de vehículos se utilizan tecnologías de Localización Automática de Vehículos o Identificación Automática de Vehículos, de este modo se determina con precisión en que punto de la ruta se encuentra un ómnibus [6]. Esa información es procesada con software de Sistemas de Información Geográfica y técnicas de posicionamiento en mapas para determinar el recorrido que el ómnibus está haciendo y si lo está haciendo a tiempo. Por otro lado la flota de ómnibus es equipada con distintos dispositivos que: verifican el correcto funcionamiento del vehículo, cuentan la cantidad de pasajeros que hay y cobran automáticamente el boleto. Estos dispositivos informan a las centrales utilizando la infraestructura de comunicación disponible. Esa información también llega a los usuarios mediante los Sistemas Avanzados de Información a Viajeros. Otro tema importante en esta área es el de la automatización del cobro de los boletos en los ómnibus. La intención es que el sistema de cobro sea más ágil para el usuario, reduzca la posibilidad de fraudes y disminuya los costos [9]. Para esto se instalan en el ómnibus máquinas automáticas que cobran el boleto.

2.4.6 Sistemas Rurales de Transporte Inteligente.

Los Sistemas Rurales de Transporte Inteligente extienden la potencialidad de los STI al medio rural. En las áreas urbanas los STI están dedicados principalmente a balancear la capacidad del sistema de transporte, pero en el área rural, eso no es un problema tan grave. En el medio rural el objetivo de los STI es principalmente monitorear la infraestructura para que el transporte sea seguro y confiable. El problema principal en el área rural son las grandes distancias, pero las tecnologías que habilitan al monitoreo remoto son de gran ayuda.

Los sistemas rurales ponen especial énfasis en mejorar la seguridad. Un estudio sobre el estado de Montana en Estados Unidos concluye que el 85 por ciento de los accidentes en las rutas podrían haberse evitado con sistemas que informarán al conductor de condiciones adversas [6]. Por eso el foco en estos sistemas esta en indicarle al conductor de condiciones riesgosas como rutas resbaladizas, intersecciones o advertencias de peligro por tomar curvas con demasiada velocidad.

Otro aspecto del que se encarga esta área es el de mejorar la productividad de los vehículos comerciales. Por ejemplo, los vehículos de transporte pueden planificar las rutas teniendo en cuenta congestiones o obras de construcción en las rutas, ahorrando tiempo y dinero. También se puede mejorar el tiempo perdido en las aduanas y controles efectuados a camiones, mediante la automatización de las inspecciones, usando sistemas de información, tecnologías de seguimiento del vehículo y bitácoras electrónicas.

2.4.7 Operación de Vehículos Comerciales.

Esta área está encargada de aplicar la tecnología de los STI al transporte de mercancías y personas. Los puntos claves en los que se espera que los STI aporten a esta área son:

- Movimiento de mercancías. En esta área se encarga de mejorar la planificación de la adquisición y distribución de mercancías. Apoyar los servicios de transporte. Automatizar la identificación y el flujo de información sobre las mercancías. Apoyar la logística, la producción justo a tiempo (just in time) y la planificación de recursos de las empresas [6].
- Operación de las empresas transportistas. Esta área se ocupa de asistir en la administración de las empresas de transporte. Algunos de los aportes son coordinar la carga de mercancías y el procesamiento de las ordenes. Controlar el despacho y el seguimiento del envío, y automatizar la administración de las flotas de vehículos, son algunos de los aportes que se espera de los STI en la eficiencia y disminución de costos de las empresas transportistas.
- Operación de los vehículos de transporte. Esta área se encarga de asistir y controlar la operación del vehículo de transporte. Algunas de las actividades son localización automática del vehículo y asistir la navegación. Monitorear el funcionamiento mecánico del vehículo y del conductor. Registrar las actividades del conductor y proveer comunicación con el centro de operaciones.
- Administración de la documentación e inspecciones automatizadas. De la administración de la documentación requerida para la operación en el transporte y realizar eficientemente las inspecciones en los puntos de control, es que se encarga esta área. Por ejemplo, las oficinas de control o aduanas identifican automáticamente al vehículo y determinan si es necesario inspeccionarlo o controlar la documentación.

Las actividades dentro del área de Operación de Vehículos Comerciales son muy variadas. Algunas de las tecnologías con las que se cuenta para lograr los propósitos planteados son:

- Información digitalizada, que permita el intercambio de datos, comercio electrónico y manejo de documentación electrónica.
- Sistemas de información, que permiten el almacenamiento, procesamiento y distribución de la información.
- Redes de comunicación.
- Tecnología para la identificación automática de mercancías, por ejemplo códigos de barra, reconocimiento óptico de caracteres e identificación por frecuencias de radio.
- Tecnologías en el vehículo que permiten obtener información del estado y las operaciones del vehículo.

2.4.8 Vehículos Inteligentes.

Los Vehículos Inteligentes son vehículos equipados con dispositivos de última tecnología que permiten asistir al conductor en la navegación. El área de Vehículos Inteligentes utiliza sensores, algoritmos inteligentes e infraestructura con la cual coopera para comprender el entorno cercano al vehículo y tomar acciones que faciliten y hagan más segura la conducción del vehículo. La cooperación con la infraestructura es en dos sentidos, es decir el vehículo recibe y envía información a la infraestructura. Por ejemplo, el vehículo le envía la posición en la que se encuentra a la infraestructura y esta le responde cual es la velocidad máxima a la que puede tomar la próxima curva sin salirse de la ruta. Las áreas más importantes en las que se trabaja son:

- Advertencia de choques. Se instalan sensores en el vehículo que determinan la posición de vehículos cercanos y obstáculos con los que se pudiera chocar. Los sensores le avisan al conductor cuando advierten alguna situación riesgosa.
- Asistencia en la conducción. Los vehículos tienen capacidad de asistir en la conducción adaptando la velocidad, frenando o advirtiéndolo si el vehículo se aparta del carril, por ejemplo si tomara una curva muy rápida y se pudiera salir del camino.
- Conducción automática. Permiten que el vehículo sea conducido automáticamente sin la necesidad de intervención humana. Distintos prototipos están siendo testeados exitosamente en Europa, Asia y Estados Unidos [10].

Uno de los problemas más difíciles de resolver en esta área es el problema conocido como el del huevo o la gallina. Es decir sin infraestructura inteligente de soporte, no es posible tener inteligencia en los vehículos y viceversa. Mientras el problema se va resolviendo, la tendencia en la industria está en potenciar principalmente la inteligencia en los vehículos [11].

2.5 Arquitectura y estándares.

El concepto subyacente de los Sistemas de Transporte Inteligentes es la integración de los distintos componentes del transporte en un sistema de sistemas. La filosofía es compartir recursos y datos para formar una red de transporte unificada. Para eso, es necesario contar con un marco que integre todos los componentes del sistema, es decir una *arquitectura del sistema*. Esta arquitectura describe en detalle que tipos de interfaces deben existir entre los componentes del STI, como intercambian información y trabajan juntos. Algunas aspectos que define la arquitectura son:

- Guías para seleccionar las estrategias de procesamiento de la información de transporte.
- Los componentes del sistema, sus funcionalidades claves y las organizaciones que deben implementarlos.
- El tipo de información que debe ser compartida.
- Un marco unificado para el diseño de los STI.

Un aspecto central que surge de la arquitectura es la interoperabilidad de los componentes. Es decir que todos los componentes del sistema deben poder aceptar, enviar y usar información y servicios generados en otros componentes. La interoperabilidad puede darse en distintos niveles como técnico, operacional e institucional. Para que esto pueda hacerse realidad se apuesta fuertemente a utilizar estándares en todos los aspectos de los STI. El desarrollo de estos estándares es un tema muy activo e importante dentro de los STI.

La arquitectura del sistema y la estandarización son dos temas por demás interesantes y complejos, pero a los efectos de este trabajo, no se profundizarán más de lo ya expuesto. Para consultarse más información se recomienda consultar las referencias bibliográficas: [6], [8] y [9] .

2.6 Sistemas de Transporte Inteligentes en el mundo.

En esta Sección se da una visión general de la situación del transporte en algunos países. No se pretende brindar un informe completo de la situación de cada país, sino una breve ilustración de las dificultades en las que se encuentran y cómo esperan resolverlos utilizando los STI. Se pretende con estos ejemplos demostrar que los STI ya son una de la realidad en algunos países y estarán más extendidos en el futuro.

Los países considerados son Japón, Francia y Estados Unidos. De este modo espera tenerse una muestra del desarrollo en el área en distintos continentes.

2.6.1 Japón.

Japón tiene aproximadamente 66 millones de automóviles, pero la red de caminos que tiene no puede soportar esa magnitud creciente de vehículos. Por lo tanto la congestión de tráfico es un problema serio que causa pérdidas económicas aproximadamente por 100 mil millones de dólares anuales, 10.000 víctimas fatales en accidentes y un daño considerable al medio ambiente. Por estos motivos, la población espera con mucha ansiedad las soluciones que puedan aportar los Sistemas de Transporte Inteligentes.

Distintos ministerios del gobierno han planificado, para el año 2015, veinte servicios al usuario entre los que se encuentran sistemas de navegación, sistemas de cobro de peajes electrónico y automatización de rutas.

Los principales tres proyectos que se desarrollan son: sistemas de comunicación de información a vehículos, sistemas electrónicos para el cobro del peajes y sistemas de automatización de rutas, en ese orden de importancia.

El sistema de comunicación de información a vehículos, llevará a los conductores información de tráfico en tiempo real. Actualmente los conductores pueden obtener información de tráfico en las tres áreas metropolitanas más importantes, Tokio, Osaka y Nagoya, así como también en 6.500 Km de rutas.

El sistema de cobro electrónico de peajes intenta aliviar las congestiones causadas por el cobro de peajes a través de 7.500 Km de rutas.

Finalmente, el sistema de automatización de rutas tiene como objetivo desarrollar un sistema seguro y práctico para asistir en la conducción o eventualmente automatizarla [12].

2.6.2 Francia.

La red de carreteras en Francia dispone aproximadamente con un millón de kilómetros, donde aproximadamente 12.000 corresponden a carreteras de alta velocidad. La seguridad en las rutas de Francia es menor que en otros países de Europa. Los accidentes de tráfico tienen un costo anual de 20 mil millones de Euros. La mayoría de los accidentes ocurren en zonas rurales y tienen como principal protagonista a camiones.

Otro problema de tráfico que se presenta, tiene que ver con la descentralización de los centros urbanos de las ciudades, siendo París la ciudad más afectada. En consecuencia las personas utilizan sus autos más que el transporte público y hacen que la cantidad de viajes en zonas periféricas se incremente considerablemente. Esto provoca congestión, problemas de polución y una reducción en el uso del transporte público que enfrenta problemas de financiación. Para atacar estos problemas, en 1997 distintas organizaciones gubernamentales, con apoyo de instituciones académicas definieron un programa de Investigación y Desarrollo llamado La Route Automatisée para resolver los distintos problemas. Se definieron cuatro áreas donde actúa el programa: mejorar la seguridad en los caminos rurales, carreteras que asistan la conducción de camiones pesados, movilidad en las zonas periféricas a los centros urbanos y mejorar el uso del transporte público [13].

2.6.3 Estados Unidos.

El tráfico en la carreteras continua creciendo a medida que aumenta la población, particularmente en áreas metropolitanas. Se estima que entre 1980 y 1999 la cantidad de millas recorridas aumentó un 76 por ciento, mientras que la construcción de caminos no ha aumentado en esa magnitud. El Instituto de Transporte de Texas estima que en año 2000 las 75 áreas urbanas más importantes experimentaron 3.6 mil millones de horas en demoras, resultando 67.5 mil millones de dólares en pérdida de productividad [14]. Por otro lado el transporte público ha aumentado alcanzando a 9.4 mil millones de viajes en el año 2000, el más alto en 40 años. Se predice que entre 1998 y 2020 aumentará el movimiento de mercancías un 69 por ciento, de 15.3 mil millones de toneladas a 25.8 mil millones de toneladas anuales. Los problemas de congestión ya no son exclusivos de las grandes ciudades y pueden encontrarse en ciudades pequeñas y zonas rurales. Con este panorama, el país tiene la necesidad de evolucionar rápidamente hacia los STI, y eso es exactamente lo que se está haciendo. En todo el país hay una gran movida hacia los STI y hay una considerable cantidad de implantaciones de distintos componentes. Pero aún no son suficientes. La intención de poder tener una infraestructura completa de STI, en los 75 centros urbanos más importantes, para el año 2005 no va a poder cumplirse satisfactoriamente [15]. De las estimaciones realizadas se espera que, el 75 por ciento de esos centro urbanos tengan un alto nivel de infraestructura de STI, mientras que el resto tengan un nivel medio. El país se encuentra seriamente comprometido con los STI y los resultados obtenidos en cada implantación de un sistema son alentadores, pero aún resta mucho camino por recorrer para llegar a los niveles esperados.

3 Sistemas Avanzados de Información a Viajeros.

Dentro de los Sistemas de Transporte Inteligentes, los Sistemas Avanzados de Información a Viajeros son componentes muy importantes, ya que interactúan directamente con los usuarios del sistema de transporte. Este componente es el encargado de llevar hasta los usuarios la información que necesiten para que puedan realizar un viaje en las mejores condiciones de eficiencia y seguridad posibles. De este modo el usuario, peatón o conductor de un vehículo tendrá contacto directo con el sistema de transporte, en forma global. En este Capítulo se presentan los Sistemas Avanzados de Información a Viajeros y se pone especial énfasis en los que se implementan mediante sitios web. Inicialmente se da una definición de los SAIV, se explican los distintos tipos que existen y cuales son sus características. Finalmente se describe el estado del arte en los SAIV.

3.1 Definición.

Los SAIV son los componentes encargados de brindar información y servicios sobre el sistema de transporte. Existen dos clases de sistema, dependiendo del tipo de información que brindan. Un tipo de sistema se encarga de la información y servicios de *tráfico*, y están orientados a los conductores de vehículos. El otro tipo de sistema se encarga del *tránsito* y esta orientado a personas que deben movilizarse utilizando el transporte público o medios alternativos como caminar o ir en bicicleta.

Para llevar la información a los usuarios los SAIV deben utilizar algún medio de comunicación, para eso hay distintas posibilidades, los medios comúnmente utilizados son televisión, radio, teléfonos, Internet o quioscos. El medio utilizado y la información consultada dependen del momento en el viaje que se encuentre la persona. Desde este punto de vista, hay tres momentos en un viaje: antes de comenzar, durante el transcurso del viaje y cuando se arriba al destino. Los SAIV deben poder llevar la información en todos los momentos del viaje.

Los SAIV deben proveer información y servicios que satisfacen múltiples propósitos, ya sean de individuos u organizaciones. Para los individuos, ya sean personas desplazándose a sus trabajos, turistas o conductores de vehículos de transporte de cargas, los SAIV implican mayor seguridad, mejor control del tiempo y menos estrés. Para las organizaciones gubernamentales encargadas de la Administración del Transporte, los SAIV son una herramienta para llevar a cabo estrategias de acción.

Todo los usuarios del sistema de transporte son los beneficiados. Algunos beneficios enumerados en el informe “Intelligent Transportation Systems Benefits and Cost. Update 2003” publicado por Mitretek Systems en el año 2003 [14], son:

- Mejoras en la movilidad, el tiempo en desplazarse de un lugar a otro disminuye.
- Optimización del sistema de transporte. Las calles son mejor utilizadas, balanceándose el uso.

- Satisfacción de los usuarios, es decir los usuarios sienten que el sistema los beneficia, ahorrándoles tiempo, dinero y brindándoles seguridad.

Por ejemplo, una persona que tenga que llegar a su trabajo podrá solicitar al SAIV que planifique la ruta, evitando así la congestión y las condiciones inseguras del tráfico matutino. El sistema de transporte público también se beneficia, por ejemplo un conductor de taxi que logra evitar demoras tendrá mejor productividad. Por otro lado, otros medios de transporte público como ser ómnibus pueden ajustar mejor el tiempo de arribo a las paradas, facilitando así la coordinación de los viajes multimodales (viajes en los que se combinan distintos medios de transporte).

Para lograr su objetivo los SAIV utilizan tecnología avanzada para recolectar información, procesarla y liberarla. La palabra ‘avanzados’ hace referencia al hecho de dejar atrás enfoques anteriores, abordados con tecnologías que finalmente no prosperaron. La información es recolectada de diversas fuentes como los Sistemas de Administración del Transporte, información climática, páginas amarillas u oficinas de turismo. Todo se almacena en forma digital, en bases de datos de computadoras. Los usuarios acceden a la información utilizando los medios de comunicación tradicionales como radio y televisión o de última tecnología como teléfonos celulares, dispositivos instalados en el vehículo o computadoras. En la Figura 3-1 se ve un esquema similar al SAIV propuesto en la ciudad de Phoenix en Arizona, Estados Unidos [6]. En la Figura 3-1 se pueden ver los distintos componentes proveedores de la información de tráfico y tránsito comunicándose con los Centros de Administración del Transporte mediante la infraestructura de comunicaciones. Luego de procesar esa información la envían a los SAIV. Estos además, la complementan con otra información también relevante para el usuario como puede ser disponibilidad de lugar para el estacionamiento del vehículo, clima o noticias. Finalmente la información llega al usuario utilizando los distintos medios de comunicación disponibles.

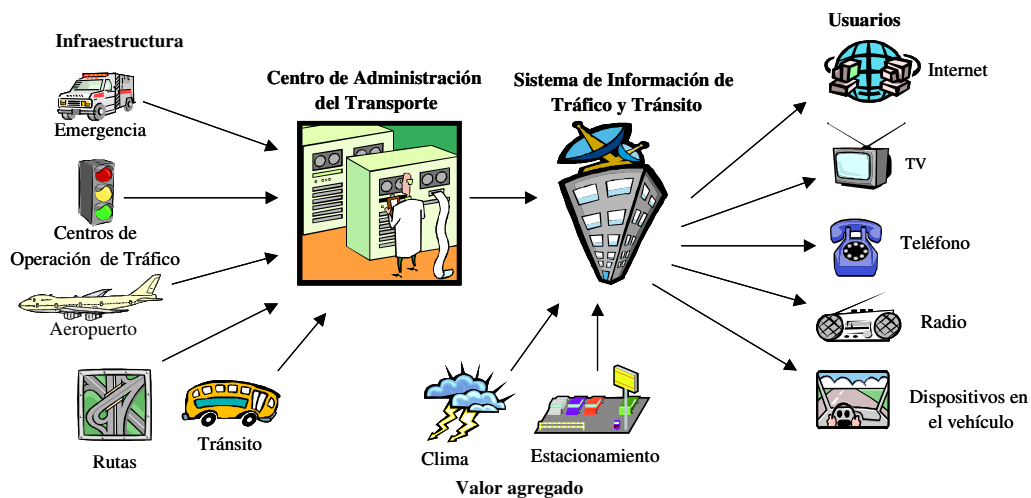


Figura 3-1. Funcionamiento de un SAIV.

3.1.1 Medios para el acceso a la información.

La forma más utilizada por la industria para clasificar a los SAIV consiste en determinar en qué momento del viaje es utilizado el servicio. Existen tres momentos en los que puede ser utilizado un SAIV: antes de iniciar el viaje, durante el transcurso del viaje o luego de finalizado. En general, cuando se consulta un SAIV antes de iniciar el viaje lo que se busca es conocer de antemano la situación de congestión de las calles buscando evitar las zonas más congestionadas y eventualmente buscar la hora más conveniente para realizar el viaje. Cuando se consulta un SAIV durante el transcurso del viaje, generalmente es porque el usuario debe cambiar la ruta que tenía planificada y requiere un nuevo camino. La consulta a un SAIV luego de que el viaje finalizó no es tratado en la literatura consultada, aunque eso seguramente se debe a que los medios sobre los que se tratan no permiten la personalización del servicio. En servicios que permiten la personalización, seguramente los usuarios demandarán este tipo de posibilidad. En las siguientes secciones se describen los medios para diseminar la información, dependiendo de la etapa del viaje.

3.1.1.1 Etapa previa al viaje.

Las plataformas utilizadas para acceder a la información en la etapa previa al viaje son principalmente de localización fija, por ejemplo:

- **Quioscos.** Los quioscos son computadoras con software de información de tráfico y tránsito, bases de datos y dispositivos de comunicación para poder acceder a información en tiempo real. Están ubicados generalmente en lugares donde hay un alto tránsito de personas, como puede ser un centro comercial o terminales de ómnibus. Cuentan con imágenes, audio, video y se operan con pantallas sensibles al tacto. Adicionalmente pueden contar con impresoras para llevar copias de la información y líneas telefónicas para comunicarse con una operadora. Los quioscos no han sido exitosos para brindar información de tráfico. Principalmente porque en los momentos en que se necesita la información, que generalmente es mientras se está conduciendo el vehículo en la ruta, los quioscos no están disponibles [2].
- **Sitios web.** Son accesibles desde cualquier lugar en donde esté disponible una conexión a Internet. La disponibilidad del acceso a Internet tanto desde los hogares, como desde el trabajo hace que esta sea la plataforma más popular para diseminar información de viajes. También es posible acceder a la información durante el viaje, utilizando conexiones inalámbricas y computadoras portátiles.
- **Televisión.** Existen canales de televisión por cable, que transmiten información de tráfico. Esta modalidad de informar es muy valorada, principalmente en los viajes que se realizan en la mañana hacia el trabajo. Los usuarios encuentran la información por televisión más útil que la de radio. Además, encuentran en la televisión un medio en el cual no es necesario hacer un gran esfuerzo de atención para recibir el mensaje [2].
- **Teléfonos con audio o texto.** Estos servicios ofrecen información mediante contestadores automáticos u operadores en línea. El usuario interactúa hablando, si el sistema reconoce voz o también digitando las opciones desde el

teclado. Actualmente Estados Unidos designó el número 511 para brindar información de tráfico y tránsito en todo el país. La implantación del servicio se está llevando a cabo con gran éxito y se espera que para el año 2005 esté disponible en todo el país [9].

3.1.1.2 Etapa en la que se está viajando.

Las opciones para recibir información mientras se está en viaje utilizan principalmente comunicaciones inalámbricas y dispositivos portátiles, por ejemplo:

- **Radio.** Algunas estaciones de radio emiten informes de tráfico, principalmente de incidentes. Son útiles, ya que es muy común que mientras se conduce, la radio permanece encendida. Sin embargo la posibilidad de extender el servicio con otras prestaciones es limitada.
- **Teléfonos celulares.** Permiten la comunicación de audio, texto y servicios de operadores en vivo. Esta opción puede tener problemas en lugares donde conducir y hablar por teléfono esté penado por la ley (como en Uruguay).
- **Computadoras portátiles.** El computador puede tener software que utilice mapas digitales de la zona para indicar una planificación de ruta. Además podría comunicarse, en forma inalámbrica con algún proveedor de información para recibir niveles de congestión o velocidades prevalecientes y brindar una planificación de la ruta que utilice esos parámetros. También puede accederse a sitios web para obtener la información. En la Figura 3-2 se observa un dispositivo incluido en los vehículos Mercedes-Benz. La compañía creó un sitio web en Internet desde la cual brinda los servicios de ruteo, mapas e información en general. Los vehículos utilizan teléfonos celulares y Asistentes Personales Digitales para acceder la información [16].



Figura 3-2. Computadoras portátiles en el vehículo.

- **Dispositivos en el vehículo.** Son dispositivos similares a computadoras con software de navegación que se instalan en el vehículo. Disponen de una pequeña pantalla donde se presenta la información y botones para la operación. Pueden utilizar componentes del vehículo especiales para la navegación como giroscopios o compases [17]. Algunos tienen capacidad para comunicarse con proveedores de información. La principal funcionalidad es asistir en la navegación. Hasta el momento no han tenido el éxito esperado, principalmente por no poder incorporar información en tiempo real en los

algoritmos de ruteo de vehículos y la resistencia de las personas a comprar dispositivos electrónicos de propósito único. El costo de poner en funcionamiento estos sistemas es alto por la inversión en infraestructura de comunicaciones necesaria [2]. En la Figura 3-3 se observa el sistema Saab que viene instalado en los modelos de lujo en vehículos General Motors [18].



Figura 3-3. Dispositivo en el vehículo.

3.1.1.3 Etapa posterior al viaje.

En la literatura consultada, a la tercer etapa no se le da mayor relevancia. Sin embargo, esta etapa es sumamente importante para el usuario ya que el llegar al destino le proporciona un momento de distensión, donde tiene la oportunidad de reflexionar sobre el viaje realizado. Algunas reflexiones o acciones que puede tomar son:

- Si la información que consultó previamente o durante al viaje fue útil y actualizada.
- Las decisiones que tomó utilizando la información fueron acertadas.
- Si el medio que utilizó para accederla es adecuado y seguro.
- Ajustar la personalización, si el sistema se lo permite.
- Reportar información que pueda servir para otros usuarios.
- Si las expectativas que tenía se cumplieron, por ejemplo la duración del viaje.

Actualmente no hay demasiada experiencia con medios de acceso a la información que permita en forma sencilla una respuesta del usuario, en ese sentido los sitios web ofrecen una oportunidad muy interesante. A medida que se extienda su uso los usuarios demandarán este tipo de oportunidades.

3.1.2 SAIV en Internet.

De todos los ejemplos expuestos, para cualquiera de las tres etapas, la opción de computadoras accediendo a sitios web tienen claras ventajas frente a las demás

propuestas. Algunas ventajas y consideraciones que se deben considerar al desarrollar un SAIV basado en sitios web accedidos por computadoras son:

- **Conexión a Internet.** Para acceder a un sitio web es necesario conectarse a Internet. La conexión puede ser mediante un cable o utilizando tecnología inalámbrica. Excepto durante el viaje, que necesariamente tiene que utilizarse una conexión inalámbrica, en el resto de las etapas puede utilizarse cualquiera de los dos métodos. A diferencia de los dispositivos en el vehículo, éste no necesita de inversiones en infraestructura de comunicación específica ya que la conexión a Internet no se implanta exclusivamente para esta aplicación.
- **Computadoras.** Las computadoras han evolucionado en distintos aspectos como velocidad de procesamiento, capacidad de almacenamiento de información o tamaño, haciéndose cada vez más pequeñas. Comparando con el dispositivo en el vehículo, la computadora ofrece muchas más opciones al usuario ya que no es un dispositivo de propósito único y el uso como herramienta de navegación sería uno más de los usos que se le da.
- **Información multimedia.** Hoy en día las computadoras tienen suficiente capacidad como para procesar una amplia variedad de información como imágenes, texto, audio, video, etc. Este tipo de información es la que comúnmente se encuentra en un sitio web.
- **Componentes de software.** Para acceder a un sitio web se requiere un software conocido como *navegador de Internet*. Los navegadores son programas capaces de descargar y presentar una página web. Son productos muy difundidos, generalmente vienen incluidos con la instalación de software de base y hay una amplia gama disponibles en forma gratuita para las distintas plataformas de computadoras.
- **Componentes de hardware.** Las computadoras tienen la capacidad de potenciarse incorporando componentes de hardware. Para el tipo de aplicaciones SAIV, componentes de localización GPS (Global Positioning System) [17] o de reconocimiento de voz, transformarían una simple computadora en un potente instrumento de navegación. Sin embargo, si no se dispone de estos componentes, una computadora igual puede ofrecer una valiosa herramienta para asistir en la navegación.
- **Interacción y personalización.** Los sitios web ofrecen una oportunidad importante en cuanto al nivel de personalización e interacción que el usuario pueda obtener, comparado con otros medios como televisión o radio.
- **Evolución.** Los sitios web pueden evolucionar, mejorando la implementación y los servicios disponibles, en forma transparente para el usuario.

Actualmente, la tecnología utilizada en Internet es suficientemente madura como para manejar los formatos de información y las aplicaciones requeridas, ya sea audio, video, imágenes, mapas digitales interactivos, aplicaciones de planificación de ruta transacciones comerciales, por ejemplo para comprar boletos de estacionamiento. Además las páginas web presentan una gran ventaja frente a otros medios de comunicación en cuanto a interfaz con el usuario y niveles de personalización.

Si se compara el uso de software preinstalado con el uso de un sitio web, el sitio web presenta algunas ventajas que lo hacen más adecuado, por ejemplo, no es necesario la instalación en el computador del usuario, se puede utilizar el servicio desde cualquier computadora que tenga acceso a Internet y un navegador. La inversión económica que

hace el usuario generalmente es menor que si comprara un paquete de software ya que en el sitio web el costo del software se amortiza entre todos los usuarios. Finalmente el usuario solo necesitará para acceder al sistema una computadora con un navegador. La situación ideal para el usuario sería si tuviera una computadora portátil y comunicación inalámbrica que pudiera utilizar mientras viaja, que es el momento donde más requiere asistencia. A medida que las computadoras de bolsillo disminuyan los costos, más personas podrán acceder a ellas y el público potencial de los sistemas aumentará.

Por lo tanto, hoy en día están dadas las condiciones tecnológicas adecuadas para que se desarrollen los sistemas de información de tráfico utilizando Internet y páginas web. Distintos sistemas prácticos e investigaciones académicas confirman esta posibilidad, por ejemplo: [19], [20], [21] y [22].

Los proveedores de sitios web son entidades públicas o privadas con distintas motivaciones. Las entidades públicas buscan principalmente controlar y utilizar todo el potencial del sistema de transporte, además de tener acceso directo a la información que recolectan. Por otro lado, las empresas privadas buscan brindar productos de valor para sus clientes y obtener beneficios económicos. Esto se refleja en la información y servicios que ofrecen cada uno de los sitios en materia de tránsito o tráfico. En los sitios públicos de tránsito se da que son completos pero la información es generalmente estática, mientras que en privados cuentan con menor funcionalidad pero se puede encontrar información en tiempo real. En cuanto a los sitios de tráfico, los sitios privados ofrecen mejores herramientas para el usuario y están seriamente comprometidos con mantener la información actualizada [3]. En las siguientes secciones se estudiarán las características y los servicios que ofrecen estos sitios web.

3.2 SAIV especializados en información de tránsito.

Los SAIV especializados en información de tránsito tienen como objetivo llevar al usuario toda la información que necesite sobre como utilizar el transporte público en forma eficiente. El aspecto más crítico de estos sistemas es lograr recopilar toda la información, de las distintas fuentes donde se genera e integrarla de tal modo, que el usuario perciba al sistema de transporte en forma global, unificado y coordinado. En la siguiente Sección se estudian los motivos por los cuales el usuario consulta la información y que servicios e información se le deben ofrecer. Luego se presentan los resultados de una evaluación sobre los servicios de los sistemas, que demuestra que aún no se brindan todos los servicios que se espera.

3.2.1 Qué servicios deben ofrecer.

Los usuarios consultan los SAIV de tránsito porque esperan: controlar el tiempo, disminuir la incertidumbre que tienen al realizar un viaje y mejorar las decisiones que realizan. Los beneficios que citan los usuarios son los siguientes [2]:

- Reducir el estrés.
- Satisfacción de utilizar el sistema.
- Mejor control sobre el tiempo y las decisiones del viaje.

La siguiente lista de herramientas e información que debe presentar un sitio web está basada en los resultados de dos trabajos publicados por el Departamento de Transporte de Estados Unidos. Una de las encuestas [2] revela el uso y las necesidades que tienen los usuarios de los sitios web o canales de televisión que brindan información de tráfico y tránsito. El otro informe [3] es un estudio sobre la información y servicios que se ofrecen en sitios web de tránsito y tráfico de organismos públicos y privados de ciudades como New York, New Jersey o San Francisco. Las publicaciones son de los primeros meses de 2000. Estas publicaciones junto con otras editadas en la misma época, [1] y [4], fueron muy influyentes en la realización de este trabajo.

- **Mapa de la ciudad.** El mapa debe presentar información turística, como ser sitios de interés, plazas, monumentos y construcciones importantes de la ciudad. Localización de las distintas oficinas públicas, seccionales de policía, bomberos y hospitales. También debe haber información sobre centros comerciales y lugares de esparcimiento. El mapa debe permitir distintos tipos de consultas, búsquedas textuales o mediante clics directos.
- **Información sobre los medios de transporte públicos.** Por ejemplo, rutas de ómnibus y subterráneos, paradas, puntos de trasbordo, horarios y tarifas.
- **Herramientas de planificación de itinerarios.** En estas herramientas, el usuario selecciona un origen y un destino y el sistema le retorna una planificación del itinerario donde le indica: qué medios de transporte debe utilizar para llegar al destino, qué línea o número de ómnibus debe tomar, en qué parada debe abordar o transbordar con otro medio. Si tuviese que caminar, en qué dirección debe hacerlo y por cuantas cuadras. Si existiesen distintas opciones, el usuario debe poder seleccionarlas e incorporarlas al itinerario. Además se debe incluir las distintas tarifas y tiempo estimado de viaje.
- **Recomendaciones sobre el mejor momento para hacer el viaje y cuales son las rutas más rápidas.** En este caso se ingresa el origen y destino y se obtienen los mejores momentos del día para efectuar el viaje y cuales son las rutas que llegan más rápido al destino. Resolver este punto requiere utilizar información obtenida en tiempo real para determinar las condiciones actuales junto con registros históricos.
- **Información en tiempo real.** Se refiere a brindar información de los ómnibus, trenes o subterráneos en tiempo real, por ejemplo su localización actual, tiempo de arribos y partidas de las paradas.
- **Comunicación con el usuario.** Habilitar direcciones de correo electrónico, números de teléfono u otros medios de comunicación para que el usuario pueda enviar preguntas o solicitar asistencia.
- **Información actualizada periódicamente.** Es importante que el usuario sepa con qué frecuencia se actualiza la información.
- **Reserva y venta de boletos en línea.** Se refiere a tener un modo seguro para comprar y reservar boletos, desde el sitio web para no tener que ir en persona hasta el punto de venta.

3.2.2 Qué servicios ofrecen.

El estudio realizado por el Departamento de Transporte de Estados Unidos da un panorama claro de los servicios que ofrecen [3]. Este estudio fue publicado a principios del año 2000 y fue tomado como base para el desarrollo de este trabajo. Si bien puede no reflejar la situación actual, debido al tiempo transcurrido desde su publicación hasta el momento en que se escribe este trabajo, de cualquier manera se expondrán sus resultados debido a la gran influencia que tuvo sobre este trabajo.

Para el informe se examinaron 85 sitios de tráfico y 120 de tránsito, en los que se evaluaba la disponibilidad de diez características predeterminadas en las dos categorías.

Los puntos que se evaluaron fueron los siguientes:

- La presencia de un mapa del sistema.
- El mapa muestra puntos de trasbordo.
- El mapa soporta consultas mediante clics.
- Información sobre autoridades de tránsito.
- Información sobre horarios y tarifas de boletos.
- Planificación de itinerarios.
- Información sobre otros modos de transporte.
- Números de teléfono y dirección de correo electrónico para que el usuario se comuniquen.
- Información en tiempo real.

Los resultados más relevantes que muestra este trabajo son:

- La mayoría de los sitios ofrecen un mapa del sistema, aunque los sitios de organismos públicos manejan además mapas regionales o de áreas particulares.
- La mayoría de los sitios no presenta información de paradas y tampoco puntos de trasbordo.
- Solo un tercio de los sitios ofrece mapas interactivos donde se puede hacer clic en él y obtener información detallada.
- La información sobre horarios y tarifas de boletos está ampliamente disponible, tanto en sitios públicos como en privados.
- Los servicios de planificación de itinerarios son muy escasos.
- Es muy difícil encontrar sitios que publiquen información en tiempo real.

Si bien el informe no incluye posibles explicaciones sobre estos resultados, se puede decir que la mayoría de los sitios están lejos de ofrecer un conjunto completo de información y herramientas. Por ejemplo, si bien la mayoría de los sitios cuenta con mapas del sistema, no ofrecen la posibilidad de interactuar mediante clics, que es la forma más natural de utilizar un mapa digital.

La información no está completamente integrada, principalmente entre los distintos medios de transporte, eso se puede ver claramente en la dificultad de encontrar los puntos de trasbordo y paradas. La falta de información en tiempo real seguramente se

debe al costo de obtenerla y publicarla, aunque el informe no detalla una razón al respecto.

Servicios como el de planificación de itinerarios, son escasos, esto es razonable si se tiene en cuenta que se basan en la disponibilidad de información completa, integrada y actualizada.

En conclusión, los resultados de esta evaluación indican que la información no es completa y las herramientas aún no están suficientemente desarrolladas.

3.3 SAIV especializados en información de tráfico.

Los SAIV especializados en información de tráfico tienen como objetivo brindarle al conductor de un vehículo información para que pueda realizar el viaje en las mejores condiciones posibles.

3.3.1 Por qué consultar información de tráfico.

Los usuarios consultan estos servicios por razones similares a las que consultan un SAIV de tránsito, reducir la incertidumbre en el viaje, disminuir las demoras por la congestión y poder controlar el tiempo [1]. También existen otras razones importantes que tienen que ver con la seguridad, la calidad de vida y el costo, monetario y ambiental que tienen las congestiones de vehículos. La demanda de información de tráfico depende de distintos factores, por ejemplo:

- Las características locales de tráfico. Incluyen atributos de la región como, el tamaño de la zona urbana y los niveles de congestión diarios. En las regiones donde existe más congestión es donde se genera la mayor demanda de información.
- La calidad del servicio de información. Este punto es tan importante como el anterior, ya que determinará la frecuencia en las consultas, el nivel de confianza y si es útil para el usuario y cumple con sus expectativas.
- Las características del viaje. El propósito del viaje y la relación que hay entre la duración del viaje y la demora por la congestión, son motivos para consultar. También si existiera flexibilidad en el momento de partida, se consultará para conocer cual es el momento más adecuado.
- Las características del conductor. Dependiendo de la actitud y los valores del conductor, es que consultará información o preferirá, por ejemplo preguntarle a otra persona por indicaciones. Generalmente las personas más jóvenes se sienten más cómodas con el uso de la tecnología y están más predispuestas a consultar información de tráfico [4].

Los usuarios del sitio web del Departamento de Transporte del estado de Washington, Estados Unidos, consultan el sitio por las siguientes razones [1]:

1. Evitar congestiones en las rutas.
2. Evaluar los efectos de incidentes en el viaje.
3. Decidir entre rutas alternativas.
4. Estimar la duración del viaje.
5. Estimar el mejor momento de partida.

Los usuarios que utilizan los SAIV reportan que las acciones más comunes que toman al consultar la información son [1]:

- Cambiar el momento de partida.
- Cambian parte o toda la ruta del viaje, variando el largo o la duración.
- Determinan con más precisión la duración del viaje y ajustan las expectativas, por ejemplo haciendo ajustes en su agenda.

Los principales beneficios que han obtenido de utilizar el sitio web han sido:

1. Ahorro de tiempo.
2. Evitar congestiones.
3. Reducir el estrés de conducir en situaciones complicadas.
4. Evitar condiciones inseguras.

Los beneficios que reportan los usuarios son alentadores e indican que los SAIV cumplen con sus objetivos. En la siguiente Sección se definen las características que tiene que tener un sitio web que publique información de tráfico.

3.3.2 Qué servicios deben ofrecer.

La información que debe brindar un sitio web a sus usuarios es muy variada y depende principalmente de las características del tráfico local. Si bien se puede generalizar que cierta información y servicios son válidos para la gran mayoría de las ciudades, la respuesta final está en gran medida en manos de los usuarios. Por ejemplo, las vistas de cámaras pueden ser preferidas por los usuarios al servicio de planificar y guiar por una ruta, ya que las cámaras permiten al usuario ejercitar su habilidad y conocimiento para determinar las condiciones del tráfico.

Los funcionalidades principales entran en las siguientes categorías [1]:

- Servicios de Información. Incluye información turística, comercial, hoteles, etc.
- Información previa al viaje. Consiste en información útil antes de comenzar el viaje, como información climática o las condiciones del camino.
- Planificación de ruta. Este servicio recomienda las rutas más favorables para llevar a cabo el viaje. Las rutas son recomendadas en términos de recorrer la menor distancia posible o llegar al destino en el menor tiempo posible.
- Información mientras se está en viaje. Por ejemplo, advertencias de construcción, congestiones de tráfico, incidentes o desvíos.
- Notificación de emergencias. Mediante este servicio el usuario solicita ayuda. El servicio responde enviando el servicio de asistencia necesario.

Los sitios web generalmente cuentan con las herramientas que se describen a continuación.

Vistas de cámaras. Las vistas de cámaras le dan la oportunidad al usuario de juzgar por sí mismo el estado de las rutas o calles. Adicionalmente a la imagen o video, se

debe proporcionar información sobre la cámara, como localización y orientación. También la hora en que se tomo la imagen. La imagen debe ser clara, particularmente en días en que la situación climática afecta la visibilidad. La Figura 3-4 muestra la imagen de una cámara publicada en el sitio web Georgia Navigator [23]. Además de la imagen se publica la hora en que la imagen fue tomada y la dirección a la que apunta la cámara.



Figura 3-4. Vista de una cámara.

Información de incidentes. La información actualizada sobre los incidentes es muy valorada por los conductores, que la utilizan en combinación con su propia experiencia para estimar la intensidad y duración de la congestión relacionada con el incidente. También se debe informar sobre las características del incidente como localización, hora y tipo de incidente, además del impacto que tiene en la zona adyacente. La Figura 3-5 muestra una sección del mapa tomado del sitio web de Georgia Navigator [23] donde se muestra un incidente (triángulo anaranjado) y obras en construcción (triángulo amarillo). Al hacer clic sobre los iconos en el mapa se presenta la información del incidente y el impacto que tiene en el tráfico. También puede verse en la imagen otros elementos como cámaras de video (puntos azules), radares (cuadrados verdes y grises) y paneles de mensajes variables.



Figura 3-5. Mapa donde se muestran incidentes.

Medidas en tiempo real de las velocidades prevalecientes, tiempos de viaje y nivel de congestión. Las velocidades prevalecientes son estimaciones de las velocidades con las que circulan los vehículos en un determinado momento. El tiempo de viaje, es una estimación aproximada de cuanto tiempo lleva llegar desde un punto a otro utilizando una ruta determinada. Cuando se selecciona entre varias rutas, la duración del viaje, es un factor decisivo, aunque las medidas directas de velocidad son preferidas por los usuarios. Otra información útil es el nivel de congestión que hay en las calles. Los mapas son especialmente útiles y claros para presentar este tipo de información. Los mapas presentan una vista en la que las calles tienen distintos

colores dependiendo del nivel de congestión que presenten. En la Figura 3-6 se observan dos mapas tomados de un sistema de información de tráfico en la ciudad de Atenas, en Grecia [24]. En el mapa de la izquierda se puede apreciar los niveles de congestión donde los colores representan los distintos niveles (de menor a mayor los niveles son: verde, amarillo, azul, magenta y rojo). En el mapa de la derecha se presenta una aplicación en la cual se indica qué tan lejos se puede llegar en 15 minutos. Los orígenes y los destinos están predeterminados y se indican por los semáforos. En la Figura 3-6 el semáforo verde es el origen.

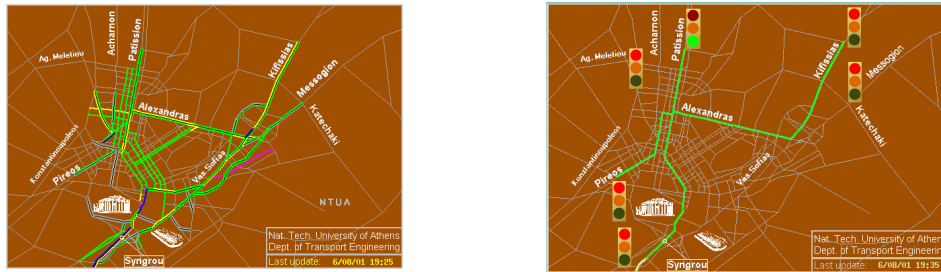


Figura 3-6. Mapas del nivel de congestión y que tan lejos se puede llegar en 15 minutos.

Servicios personalizados que tengan en cuenta las condiciones de tráfico. Por ejemplo, planificar la ruta del usuario. Este servicio consiste en indicar al viajero el camino que debe seguir hacia el destino y guiarlo. El servicio consiste en presentar un mapa con el camino a seguir e indicaciones de las calles que se toman y en las que se debe doblar. En la Figura 3-7 se puede ver el mapa y la ruta planificada por el prototipo de SAIV que se implementó para este trabajo y que se describirá en el siguiente Capítulo.

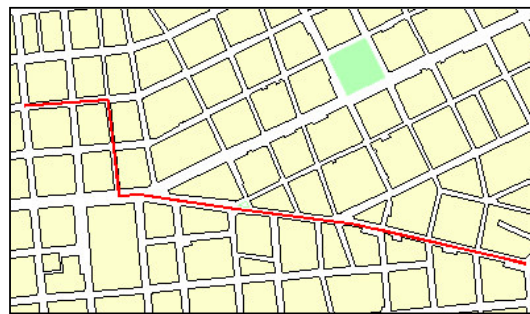


Figura 3-7. Mapa con la ruta planificada.

Condiciones climáticas. El clima es una forma de incidente y los conductores deben saber de que manera afecta al tráfico. Principalmente en lugares donde el clima puede afectar al tráfico, como nieve, heladas y lluvias. En caso de emergencias como por ejemplo tornados o inundaciones, la información de rutas de escape, es de vital importancia.

Información de rampas y accesos. Las rampas y accesos permiten acceder a las carreteras de alta velocidad, la información de la cantidad de vehículos en la cola es útil para seleccionar la rampa adecuada para ingresar a la carretera.

Información predictiva y tendencias. Se refiere a predecir las condiciones inmediatas del tráfico conjugando información actual, obtenida en tiempo real e información histórica esperada a cierta hora del día, día de la semana o condiciones climáticas. La información sobre las tendencias indica si la congestión se disipa o se agrava.

Información de estacionamiento. En muchas zonas no hay suficientes lugares de estacionamiento para cubrir la demanda en horas pico. Por lo tanto si se informa qué zonas de estacionamiento tienen lugar disponible, se evita tener que buscar lugar conduciendo.

Información de paneles de mensajes variables y radares. Los paneles de mensajes variables son paneles electrónicos colocados en distintas partes de las rutas que presentan información relevante para el conductor, por ejemplo las condiciones de tráfico o advertencias de accidentes. El mensaje es controlado por el Centro de Administración del Transporte, que lo cambia según lo crea conveniente. Otra información de utilidad, es la información recolectada por radares y otro tipo de sensores disponibles. La Figura 3-8 muestra los mensajes publicados en el sitio web de Georgia Navigator [23]. La Figura de la izquierda muestra el mensaje presentado en un panel y el de la derecha la información obtenida de un radar. El mensaje completo del panel no se ve en la Figura 3-8 pero dice: “Tiempo de viaje hacia la calle 10 por la salida 84, menos de 7 minutos”. La información del radar indica que la velocidad promedio es superior a 50 millas por hora. Esta información puede accederse haciendo clic desde el mapa del sistema que presenta iconos con la localización de los paneles y los radares (véase la Figura 3-5).



Figura 3-8. Mensaje en un panel e información de un radar.

Otras características que deben tener los SAIV [1]:

Cubrimiento. El servicio debería cubrir todas las rutas mayores y de alta velocidad. Además debe brindar información sobre carriles especiales por ejemplo para vehículos pesados. Los caminos con mayores dificultades deben ser cubiertos.

Frecuencia de actualización de la información. La actualización de la información debe ser tal que refleje las condiciones actuales del tráfico.

Información móvil. Los conductores necesitan la información de tráfico mientras conducen. En muchos viajes, la información que tuvieron al iniciar el viaje, está obsoleta al llegar a un lugar donde deben desviarse. En esta situación es donde existe la mayor demanda por los servicios de información. Los servicios deben tener distintas aplicaciones dependiendo de si el sistema es accedido desde la ruta o no.

Interfaz con el usuario y características operativas. Las interfaz y las características operativas dependen del medio que se utilice para informar. Los sitios web están beneficiados por las posibilidad de presentar distintos tipos de información, pero son sensibles a la velocidad de la computadoras y la conexión a Internet. Los servicios telefónicos requieren de acceso rápido y sencillo a la información, ya que se utilizan frecuentemente desde el camino. Los servicios que utilizan reconocimiento de voz mejoran el acceso a la información sobre los que se utilizan mediante los botones del teclado.

3.3.3 Qué servicios ofrecen.

El estudio mencionado anteriormente [3], muestra los siguientes resultados:

- Dos tercios de los sitios no proveen información en tiempo real.
- La información sobre incidentes es abundante. Aunque en sitios de organismos públicos solo se presentan advertencias de construcción, más que información de incidentes en tiempo real.
- Muy pocos sitios ofrecen información sobre las velocidades prevalecientes e imágenes de cámaras en tiempo real.
- La mitad de los sitios ofrece información sobre otros modos de transporte y programas como servicios de reparación mecánica de vehículos.
- Muy pocos sitios ofrecen servicios de valor agregado al usuario como, planificar la ruta, alertas de tráfico personalizadas o la duración estimada de un viaje en tiempo real.
- Dos tercios de los sitios actualiza la información cada 10 minutos o menos.
- Los mapas son ampliamente utilizados, principalmente los que muestran el área completa. Aunque poco más de la mitad de los sitios ofrece mapas interactivos.

Para los sistemas de información de tráfico, la información en tiempo real del estado de las calles es una herramienta básica, a partir de la cual, se pueden construir servicios que la procesen y agreguen valor para el usuario. Los resultados del estudio [3] indican que la mayoría de los sitios no dispone de información en tiempo real y por lo tanto tampoco pueden tener aplicaciones avanzadas como planificar rutas, información de las velocidades prevalecientes, imágenes de cámaras o alertas de tráfico. En el informe no se detalla una razón de esto, pero puede deberse al alto costo de obtener la información, procesarla y ponerla en línea o simplemente porque no hay infraestructura disponible.

Por otro lado existe un importante compromiso por mantener la información actualizada ya que un número importante de sitios actualiza la información cada 10 minutos o menos.

3.3.4 Usuarios de los SAIV.

En esta Sección se presentan las características de los usuarios de los SAIV y cual es su nivel de aceptación.

3.3.4.1 ¿Quiénes son los usuarios?

Los usuarios de los SAIV son, principalmente personas que tienen un empleo y que pierden mucho tiempo en el viaje entre sus hogares y su lugar de trabajo. Los usuarios que experimentan más congestión en sus rutas son los que más frecuentemente utilizan los SAIV. En Seattle, los usuarios frecuentes de los sitios web reportan que dos tercios del viaje entre sus hogares y el lugar de trabajo es debido a la congestión [4]. Los usuarios de los servicios que se emiten por televisión reportan resultados similares.

Un estudio de mercado [4] que segmentó a los usuarios basándose en sus actitudes, necesidad de control, valor del tiempo, preferencias personales de viaje y uso en las tecnologías avanzadas y determinó que existen tres grupos principales de usuarios. Estos grupos fueron llamados: *buscadores de control* (control seekers), *usuarios enfocados en el web* (webheads) y *buscadores de información previa al viaje* (pre-trip information seekers).

Los buscadores de control se caracterizan por utilizar ampliamente dispositivos portátiles avanzados como teléfonos celulares y computadoras portátiles. Estos usuarios utilizan los SAIV para cumplir su agenda, mantenerse informados, ahorrar tiempo y ser eficientes.

El grupo de usuarios enfocados en el web está formado por personas que utilizan fuertemente la tecnología más innovadora, principalmente computadoras e Internet. A diferencia del grupo anterior, estos usuarios son fanáticos de la tecnología en sí misma y la ven como algo más que una herramienta para potenciar la capacidad de las personas. Este grupo de usuarios está interesado principalmente en acceder mediante Internet al servicio, otras tecnologías alternativas como la televisión no le resultan atractivas.

El tercer grupo de usuarios, los buscadores de información previa al viaje, son los que están menos predispuestos a utilizar nuevas tecnologías. Tienen una baja aceptación de Internet y desconfían de las computadoras, por lo que utilizan los servicios que difunden la información por televisión. El grupo está formado por personas de mayor edad que el resto de los grupos.

3.3.4.2 Aceptación de SAIV.

La aceptación de los SAIV es aún muy baja con respecto a la aceptación esperada [9]. Según un informe [1] esto puede explicarse si se observa que para los conductores la principal referencia que tienen sobre las condiciones del tráfico es su propia experiencia. Generalmente evalúan su habilidad para predecir las condiciones de tráfico como buena y opinan que otros medios, como información por radio, no es confiable. Como resultado de esto los conductores piensan que no es posible obtener información de tráfico precisa y útil. Otros conductores piensan que no hay alternativa a las congestiones de tráfico y por lo tanto no hay razón para utilizar un servicio de información. A pesar de estas opiniones, tienen una gran expectativa por los SAIV, principalmente por los sistemas que utilizan sitios web.

Es muy probable que en los lugares donde existe más descreimiento, seguramente funcione mejor la información que permite que el usuario pueda evaluar por sí mismo el estado de las calles, por ejemplo las vistas de cámaras y las velocidades prevalecientes tengan buena aceptación.

3.4 Información y Datos.

Un aspecto fundamental para los SAIV es la información que puede recolectar él mismo o le pueden enviar otros componentes del STI. Recolectar información útil, precisa, confiable y a tiempo no es una tarea trivial. El tema es suficientemente complejo como para que se creara un componente específicamente dentro de los STI. En la arquitectura nacional de Estados Unidos se definió en 1999 un componente llamado Servicio de Archivo de Datos de Usuarios que se encarga de todos los aspectos relacionados con los datos [6]. Este componente se encarga del almacenamiento, control de calidad y de dejar disponible los datos para otras aplicaciones, entre otras cosas.

El tratamiento de datos en el SAIV tiene al menos estos tres pasos [25]:

1. **Recolección de datos.** Este paso implica recolectar datos relevantes que influyen en las condiciones del viaje.
2. **Fusión de datos.** En este paso se deben consolidar los datos obtenidos de distintas fuentes para que queden integrados de modo que pueden utilizarse. Este proceso es complicado ya que deben identificarse posibles conflictos que pueden ocurrir con los datos obtenidos de distintas fuentes. Se debe obtener una visión unificada de las condiciones actuales.
3. **Diseminar datos.** Este paso implica comunicar los datos fusionados al consumidor cuando, como y donde los requiera.

Algunos aspectos que hay que tener en cuenta durante el tratamiento de los datos en los pasos anteriores son:

- **Contenido de los datos.** Es necesario determinar que información se recolectará sobre los distintos eventos que tienen relevancia. Por ejemplo de un accidente de tráfico se pueden recolectar: detalles como localización o gravedad, hora en que ocurrió e impacto en el funcionamiento normal. Hay

documentos que guían que tipo de información se debe recolectar para cada tipo de dato [25].

- Calidad y control de los datos. Las fuentes de información son muy variadas y la información que proveen también. Es entonces que es importante controlar que los datos sean correctos y tengan los niveles de calidad que este tipo de aplicaciones necesita. Por ejemplo, en [6] se menciona que unos detectores de tráfico en San Antonio reportaron buenos datos el 76,5% del tiempo, el 1% fue sospechoso y el 22,5 % de los datos se perdieron. Por tanto las fuentes de datos deben ser comprendidas e inspeccionadas para determinar si existen errores y corregirlos.
- Administración y almacenamiento. Los STI recolectan datos las 24 horas del día, los 365 días del año. Esto puede determinar que la cantidad de datos que se manejan alcance rápidamente niveles inmanejables. Por ejemplo, el sistema PeMs desarrollado por la Universidad de Berkley para almacenar información de California y Florida recolecta diariamente 2 Giga bytes de información que proviene de detectores, incidentes y clima [9]. En estas condiciones es necesario tener al menos una importante infraestructura de equipos de computadoras, gestores de bases de datos, personal especializado, para poder manejar estos volúmenes de información.
- Análisis. Para lograr un correcto análisis de la información, es necesario transformar la gran cantidad de datos existente en información de valor. Para eso es necesario contar con herramientas automáticas que procesen los datos y generen información que pueda ser analizada y entendida por las personas.
- Acceso, propiedad y privacidad. Este puntos es especialmente delicado ya que generalmente los datos recolectados son utilizados por distintos componentes del STI o por empresas privadas como proveedores privados de información a viajeros.

El manejo de datos es un aspecto extremadamente sensible de los SAIV y los STI. Mantener el control, la calidad y el volumen de los datos es una tarea que debe ser completamente automatizada. El volumen de datos hace prácticamente imposible que el trabajo sea desarrollado en forma manual. Para los STI un correcto manejo de los datos es un pilar fundamental sobre el que están basados todos los componentes del sistema de transporte y en particular los SAIV.

3.5 Estado del arte en Sistemas Avanzados de Información a Viajeros.

Durante la década de los 90 comienza el auge del área de los Sistema de Información a Viajeros. Principalmente se realizan implantaciones experimentales de sistemas que utilizan dispositivos en el vehículo. Por ejemplo Pathfinder [26] es desarrollado entre 1991 y 1992 en una carretera de Santa Mónica en Los Angeles, Estados Unidos. El sistema tenía tres componentes: un centro de operaciones con computadoras que procesaban la información de tráfico, un sistema de comunicación de radio que enviaba cada un minuto la información a los vehículos y un sistema en el vehículo que consistía en una computadora que presentaba el mapa de la zona con el vehículo e indicaba las áreas más congestionadas de la ruta.

El desarrollo de Internet a mediados de los años 90, abre la posibilidad de que se utilice este medio para diseminar la información de tráfico y tránsito. También el desarrollo de las comunicaciones inalámbricas y las computadoras de bolsillo, permiten que en todo momento se pueda tener la información disponible. La conjunción de estos avances hace que el área se expanda creándose un frente de trabajo nuevo, el de crear sistemas y aplicaciones que utilizando Internet y computadoras móviles puedan llevar información y aplicaciones de tráfico y tránsito. Esta área despierta el interés inmediato en la comunidad académica, que tiene un fuerte impulso durante finales de los 90. Distintas universidades realizan proyectos en esta área y otras áreas del transporte. Por ejemplo PULSAR en la Universidad de Princeton [27], DRGS en la Universidad de Waterloo [28], DynaMIT en el Massachusetts Institute of Technology [29] o Transims en Los Alamos National Laboratory [30].

Hacia principios del año 2000 son publicados distintos reportes sobre los Sistemas Avanzados de Información a Viajeros que reflejan distintos aspectos de la situación de estos sistemas en ese momento. El reporte “Who are ATIS Customers?” [4] describe una segmentación de los clientes dependiendo de sus valores y actitudes frente a la información de tráfico y tránsito. Simultáneamente se publica “What do ATIS customers want?” [1] que enumera distintas características que tienen algunos servicios de información que han sido acertadas. También se publica “Features of Traffic and Transit Internet Sites” [3] que muestra los resultados de una encuesta llevada a cabo en 85 sitios con información de tráfico y 120 con información de tránsito. Los resultados de esta encuesta demuestran que la situación general de los sitios públicos y privados, en ese momento, es de muy poco desarrollo y que los sitios aún no alcanzan el nivel esperado. Sobre finales del año 2000 se publica “What have we learned about ITS” [2] donde se resume los avances realizados al cumplirse 10 años del inicio de la aplicación de los STI en Estados Unidos. Algunas de las conclusiones a las que se llegan en el informe son: aún no está claro que los SAIV puedan ser emprendimientos comerciales exitosos, Internet será la mayor fuente de información para los viajeros y la integración de la información de tráfico con los Sistemas de Administración del Transporte es prácticamente inexistente.

Esta serie de reportes junto con otros anteriores [31] tratan de impulsar el desarrollo de los SAIV a nivel privado. Dentro de los Sistemas de Transporte Inteligentes, los SAIV es una de las áreas donde se espera una mayor participación de las empresas privadas. A pesar de esto el sector público hoy en día sigue siendo una muy importante fuente de información.

Actualmente el interés académico por los SAIV parece haber disminuido. Los grandes proyectos iniciados a mediados de los años 90 en las universidades, ya han concluido. El esfuerzo ahora se concentra en otras áreas del transporte como la cooperación entre vehículos [32], asistencia al conductor desde el vehículo [33] o el problema conocido como el problema del huevo o la gallina, que consiste en determinar qué infraestructura y qué dispositivos se deben poner en el vehículo para que puedan cooperar [11]. El problema radica en que si no se tiene la infraestructura entonces tampoco se pueden desarrollar los dispositivos en el vehículo y viceversa. El desarrollo actual de los SAIV está en manos de la industria y las autoridades de tráfico. Sin embargo el desarrollo de los SAIV no está detenido, a medida que surgen nuevas tecnologías, estas son rápidamente asimiladas [34].

3.5.1 Tecnologías utilizadas actualmente.

Existen varias formas de categorizar la información a viajeros, la industria ha tomado como estándar clasificarla dependiendo de la etapa en el viaje en que es recibida. Existen dos etapas en las que se recibe la información, en la etapa previa a iniciar el viaje y mientras se está en la ruta.

En la etapa previa al viaje se consulta información como: cual es el mejor momento de partida o la ruta a seguir. Para eso se utilizan principalmente medios de localización fija, como puede ser televisión, computadoras con acceso a Internet, teléfonos o quioscos. Todos estos sistemas tienen sus propias características y sus ventajas y desventajas. Las computadoras son los métodos más novedosos y aún se continúa desarrollando todo el potencial que pueden alcanzar para las aplicaciones SAIV. Hoy en día los sitios web básicamente informan de las condiciones del tráfico en tiempo real, pero no proveen servicios como el de planificación de recorridos.

En cuanto a la información que se accede cuando se está en la ruta hay básicamente tres medios que se están utilizando actualmente: teléfonos inalámbricos, dispositivos en el vehículos y Asistentes Personales Digitales (APD). No está claro cual de estos tres medios prevalecerá sobre los demás en el futuro. Los tres están siendo fuertemente impulsados. En el área de la comunicación utilizando teléfonos, Estados Unidos está desarrollando un programa en el cual estandarizó el número 511 para brindar información de tráfico. Se espera que para el 2005 el programa esté desarrollado en todo el país [9].

La Telemática es otra área que juega un rol importante para los SAIV. Se refiere a productos, servicios e infraestructura que envía información, comunicación y entretenimiento a los dispositivos en el vehículo o dispositivos móviles [9]. Además de los dispositivos en el vehículo, que pueden estar incluidos en el vehículo de fábrica o pueden comprarse por separado, se ofrecen los servicios de información que recibe el vehículo. El desafío histórico de la telemática siempre fue que los clientes pagaran por ese servicio de información. Aunque en Estados Unidos las dos compañías más grandes de servicios de telemática ya son rentables económicamente [35]. Actualmente la disponibilidad de los dispositivos en el vehículo ha crecido en los últimos años. Actualmente 200 modelos de vehículos que representan 32 marcas pueden ser adquiridos con dispositivos de telemática incluidos. En el año 2002 se estima que aproximadamente fueron vendidos 2.5 millones de vehículos con dispositivos de telemática disponible. Con la infraestructura instalada hoy en día, Estados Unidos espera alcanzar los 100 millones de vehículos para el 2010 [36]. Con estas estimaciones resulta claro que la telemática está en pleno auge y será extremadamente popular a medida que los vehículos tengan incluido dispositivos de fábrica.

Los Asistentes Personales Digitales por otro lado podrían incluirse como parte de la telemática pero en realidad tienen una gran diferencia, no son dispositivos de propósito único. Generalmente las personas se resisten a comprar dispositivos electrónicos de propósito único. Un APD es un serio competidor de los dispositivos en el vehículo que ofrece la telemática. Principalmente en lugares donde no existe la

infraestructura requerida por un producto de telemática o los dispositivos en el vehículo pero si existe la infraestructura para conectar a Internet un APD en forma inalámbrica, como es el caso en Uruguay. De hecho ya algunas marcas de vehículos incorporan dispositivos en el vehículo que son APD [18]. Los APD acceden a sitios web que brindan la información y servicios de tráfico necesarios. Desde un punto de vista funcional son similares, aunque algunos dispositivos en el vehículo están mejor integrados con el vehículo y pueden brindar información sobre el funcionamiento del vehículo.

Es probable que en el futuro estas tres tecnologías convivan juntas y cada cual tenga su propio desarrollo.

4 Un prototipo de SAIV de tráfico.

4.1 Introducción.

En este Capítulo se describe la implementación de un prototipo de un Sistema de Avanzado de Información a Viajeros de tráfico, utilizando un sitio web. La implementación de un SAIV con información de tránsito podría decirse que es similar a la de uno de tráfico, los módulos necesarios son los mismos. La diferencia más grande está en el tipo de información que se procesa y los algoritmos que se utilizan para planificar las rutas. Como originalmente este trabajo está orientado hacia los sistemas de tráfico, no se profundizará en el desarrollo de un sistema de tránsito. Pero la arquitectura e implementación que se utiliza para el desarrollo del sistema de tráfico, puede ser reutilizada en un sistema de tránsito.

La motivación principal para el desarrollo del prototipo es realizar un aporte más al tema ya que aún es muy nuevo. Los objetivos de realizar un prototipo son principalmente:

- Evaluar si la falta de desarrollo de los SAIV se debe a dificultades en la implementación.
- Evaluar si existen herramientas adecuadas para cada uno de los componentes o es necesario desarrollarlas.
- Acercar la tecnología SAIV al Uruguay utilizando información y características de la ciudad de Montevideo.

4.2 Arquitectura genérica de un SAIV.

Los SAIV deben brindarle a los usuarios información de tráfico y aplicaciones. Si bien en el Capítulo anterior se mencionaron algunas aplicaciones, estos sistemas están aún en una etapa de evolución, por lo que es probable que surjan nuevas aplicaciones para incorporar. Por lo tanto el diseño del sistema debe ser suficientemente flexible como para permitir que se implementen los requerimientos descritos en el Capítulo anterior y además para adaptarse a nuevos requerimientos que puedan surgir. Debido a la complejidad del sistema, es necesario dividirlo en distintos módulos que implementen funcionalidades específicas. La integración de todos esos módulos formará el SAIV. En la Figura 4-1 se propone una arquitectura genérica de un SAIV, que utiliza un sitio web como interfaz con el usuario. Los módulos están representados por los rectángulos y las flechas indican que un módulo utiliza servicios de otro (en la dirección de la flecha).

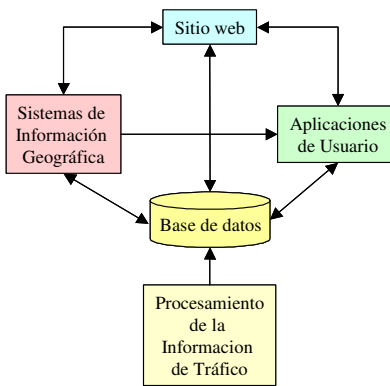


Figura 4-1. Arquitectura genérica de un SAIV.

En el centro de la arquitectura está la Base de Datos, que es donde se almacena toda la información del sistema. La Base de Datos es el pilar fundamental del sistema y donde están basados todos los módulos. La Base de Datos interactúa con todos los módulos del sistema. Algunos de ellos, como puede ser el módulo de Procesamiento de la Información de Tráfico, solo almacenan información para que el resto de los módulos la usen. Otros como el módulo de Aplicaciones de Usuario utiliza y genera información en la Base de Datos.

El módulo de Procesamiento de la Información de Tráfico, es el encargado de recibir la información de tráfico, procesarla e ingresarla en la Base de Datos. El requerimiento más importante de este módulo es el de procesar la información en tiempo real que envían los sensores o los Centros de Administración del Transporte. Los sensores que se describen en [20] por ejemplo, envían 30 Kbytes de información por minuto. El módulo consiste básicamente en un proceso demonio, que continuamente está procesando información. El procesamiento consiste en recibir la información, transformarla desde el formato de llegada al formato en que va a ser almacenada en la Base de Datos. También puede ser necesario efectuar algún tipo de computo, antes de su almacenamiento.

El componente de información geográfica es esencial para el sistema. Este componente se encarga de todo el manejo relacionado con la información geográfica. Para eso la Base de Datos debe tener capacidad para almacenar datos geográficos. El componente de Información Geográfica es el encargado de presentar y manejar mapas, administrar la información geográfica y georeferenciar la información de incidentes.

El módulo de Aplicaciones de Usuario implementa servicios que utiliza el usuario. Por ejemplo los algoritmos de planificación de rutas. El módulo utiliza la información almacenada en la Base de Datos y también puede necesitar servicios del módulo de Información Geográfica, ya sea para realizar operaciones de procesamiento o para la presentación.

Finalmente el último módulo implementa el Sitio web. Esto provee la interfaz con el usuario, desde donde accede a toda la información y servicios del SAIV. El módulo del Sitio web interactúa con la Base de Datos de donde obtiene la información de configuración, con el módulo de Información Geográfica que le brinda mapas y con el módulo de Aplicaciones de Usuario ejecutando los algoritmos de planificación de rutas. El módulo del Sitio web debe prestar especial atención a los aspectos de interacción con el usuario. Las secciones que brindan información para los usuarios que están en el viaje, deben tener en cuenta que el usuario está enfocando su atención en conducir y deben permitir el acceso y la presentación de la información de manera simple y clara.

4.3 Arquitectura del prototipo.

El diseño del prototipo, sigue la arquitectura propuesta en la Sección anterior. La Figura 4-2 muestra los distintos módulos y su interacción.

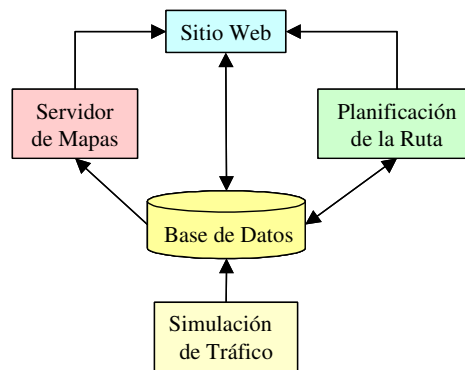


Figura 4-2. Módulos del prototipo.

En la Base de Datos se almacena toda la información, ya sean mapas digitales, información de usuarios del sitio web o información de tráfico. El módulo de Planificación de Rutas, se encarga de ejecutar los algoritmos de planificación. El Servidor de Mapas provee los mapas que se le presentan al usuario. El módulo de Simulación de Tráfico se encarga de generar, procesar y almacenar en la Base de Datos las condiciones del tráfico. Como no se cuenta con ese tipo de información, entonces el simulador se encarga de generarla. El módulo del Sitio Web, llamado Viajero Informado, implementa la interfase con el usuario. En las siguientes secciones se detalla el uso y la implementación de cada uno de los módulos. El módulo de información de tráfico se describe en el siguiente Capítulo.

4.4 Base de Datos.

La Base de Datos es el punto más importante del sistema. Cumple dos funciones, almacenar toda la información del sistema y comunicar a los distintos módulos ya que algunos dejan información disponible para que otros la utilicen. La Base de Datos que se utilice para implementar un SAIV debe cumplir con ciertos requerimientos que la

hagan soportar las dimensiones que tiene el sistema. Algunos de los requerimientos que se le deben exigir a la Base de Datos para que sea adecuada son:

- Soporte para información geográfica. Generalmente, los productos de Base de Datos no soportan nativamente el manejo de este tipo de información, pero es posible extenderlos con componentes especiales. La información geográfica es fundamental en el desarrollo del sistema ya que mucha de la información utilizada será presentada en mapas. Información como calles, manzanas, plazas, ubicación de hospitales o cuarteles de bomberos, es el tipo de información que debe tener representación geográfica.
- El Gestor de Base de Datos debe soportar manejo de transacciones, acceso concurrente de usuario, soporte eficiente para gran cantidad de registros, herramientas de administración, entre otras propiedades deseables para el tipo de aplicaciones que es un SAIV y un sitio web. Principalmente deben exigirse las propiedades de atomicidad, consistencia, aislamiento y durabilidad que el Gestor debe tener para lograr un correcto manejo de transacciones [37].

En la etapa inicial del proyecto se seleccionó para manejar la información geográfica archivos Shapefile [38] y como gestor de Base de Datos SqlServer de Microsoft. Esta combinación de tecnologías resulta natural ya que la información geográfica estaba originalmente en ese formato y no se disponía de otra alternativa. Esta combinación presenta algunas dificultades para su uso. Por un lado es necesario contar con una biblioteca de funciones capaces de poder operar con la información geográfica en el formato Shapefile. Por otro lado los objetos geográficos, almacenados en los archivos, deben hacer referencia a información almacenada en la Base de Datos. Esto hace que sea necesario controlar la integridad de los datos mediante programas.

Afortunadamente, durante el transcurso de este proyecto surge una nueva opción tecnológica. Esta nueva posibilidad esta dada por el gestor de Base de Datos llamado PostgreSQL que es de código abierto (open source), que cumple con las características exigidas anteriormente [39]. Para manejar información geográfica se utilizó una extensión llamada PostGIS que permite almacenar objetos geográficos, crear índices especiales sobre los datos geográficos y realizar operaciones básicas sobre los objetos geográficos.

4.4.1 PostgreSQL.

PostgreSQL es un gestor de Base de Datos de código abierto desarrollado a partir del gestor Posgres desarrollado en la Universidad de California. Si bien no es un gestor orientado a objetos ya que utiliza el modelo relacional, incluye algunas características de la teoría de objetos. Aunque PostgreSQL nació como un proyecto de investigación, hoy en día esta catalogado como el gestor de Base de Datos de código abierto más avanzado. Ya es considerado por la industria como un gestor de Base de Datos que está alcanzando los niveles que tienen otros gestores comerciales, especialmente en combinación con el sistema operativo Linux. Está disponible para múltiples sistemas operativos como Linux, Windows y distintas versiones de Unix y plataformas de hardware [39].

4.4.2 PostGIS.

PostGIS es una extensión de PostgreSQL de libre uso, desarrollado por la empresa canadiense Refrations Research Inc. [40]. Implementa un estándar para el manejo de información geográfica en una Base de Datos. La especificación del estándar fue creada por un consorcio de empresas llamado Open GIS Consortium [41], cuya intención es promover la estandarización dentro del área de los sistemas de información geográfica. La falta de estándares ha sido un problema dentro del área de los Sistemas de Información Geográfica, que perjudica la interoperabilidad de los sistemas y por lo tanto el desarrollo del área [42]. El estándar que implementa PostGIS es el “Simple Features Specification for SQL” [43]. Este estándar extiende el Lenguaje Estructurado de Consultas (SQL) con operaciones para manipular información geográfica, como algo nativo del lenguaje. Para eso extiende los tipos de datos básicos con nuevos tipos geográficos como puntos, líneas o polígonos, define el formato en que deben ser almacenados y las operaciones para manipularlos. A las sentencias del lenguaje se le agrega comandos para crear objetos, tablas, índices y utilizar operadores geográficos.

4.4.3 Organización de la Base de Datos.

La Base de Datos tiene varias tablas que son utilizadas por los distintos módulos para almacenar la información. El sitio web utiliza algunas tablas para almacenar información administrativa sobre el manejo de los datos geográficos y sobre el usuario. Por ejemplo, cuando el usuario ingresa al sistema se crea información sobre la sesión donde se almacenan parámetros relacionados con el manejo de la información geográfica que esta utilizando.

Otras tablas almacenan información geográfica. Como la información originalmente estaba en los archivos Shapefile, fue necesario convertirla para ingresarla en la Base de Datos. Para eso se utilizaron herramientas de PostGIS y se desarrollaron otras. Las herramientas desarrolladas fueron dos. Una de ellas transforma la información de calles eliminando los puntos que le dan forma a la calle y solo deja la información correspondiente a las esquinas, de este modo es más sencillo y eficiente generar el grafo de calles. También genera una capa con las esquinas (en la siguiente Sección se describen en detalle este tema). El Servidor de Mapas utiliza estas tablas para generar los mapas.

La simulación del tráfico de la ciudad utiliza las tablas con información de las calles de la ciudad. Cuando la simulación termina de ejecutar almacena la información generada en otras tablas.

Finalmente los algoritmos de planificación de rutas utilizan las tablas con información de las calles y las tablas con la información generada por la simulación para ejecutar. Cuando generaron la planificación de la ruta, la almacenan en tablas donde se guarda cada tramo de la ruta.

Las tablas en la Base de Datos fueron creadas automáticamente utilizando Genexus [44]. Las tablas que involucran información geográfica, fueron creadas utilizando directamente sentencias SQL, ya que Genexus no soporta el manejo de información geográfica. En la Sección 4.7 se dan más detalles sobre el desarrollo en Genexus.

A continuación se detalla el diseño lógico de la Base de Datos.

4.4.3.1 Diseño lógico de la Base de Datos.

En la Figura 4-3 se presenta un diagrama de Bachman [44] del diseño de la Base de Datos. En el diseño de la Base de Datos se reflejan los distintos componentes que tiene la herramienta.

El primer grupo de tablas está formado por las tablas que utiliza Viajero Informado para administrar los parámetros del sistema, los usuarios y las sesiones que estos tienen en el sitio. Algunas de estas tablas son GEDDtaM1, WebUsr, SPSesion y todas las tablas relacionadas en ese grupo. Además hay otras que no están directamente relacionadas como, Debug, LayNombreCalles, KBScripts y NumNumerador pero también son utilizadas.

Otro grupo de tablas es utilizado por los algoritmos de planificación de rutas. En las tablas EjecucionRuteo, Ruta y RutaTramos se almacenan las rutas planificadas. Estas tablas están relacionadas con las tablas que almacenan la información de las calles de la ciudad que son GeoEjes, GeoEsq, GeoEjes2pDatos.

La tabla SimLyrMedidas es utilizada por el simulador de tráfico para almacenar la información de tráfico.

Finalmente existe otro grupo de tablas que almacenan información geográfica que es utilizada por el Servidor de Mapas para construir los mapas. Estas tablas son: Manzanas, Playas, Plazas y Sep_Viales.

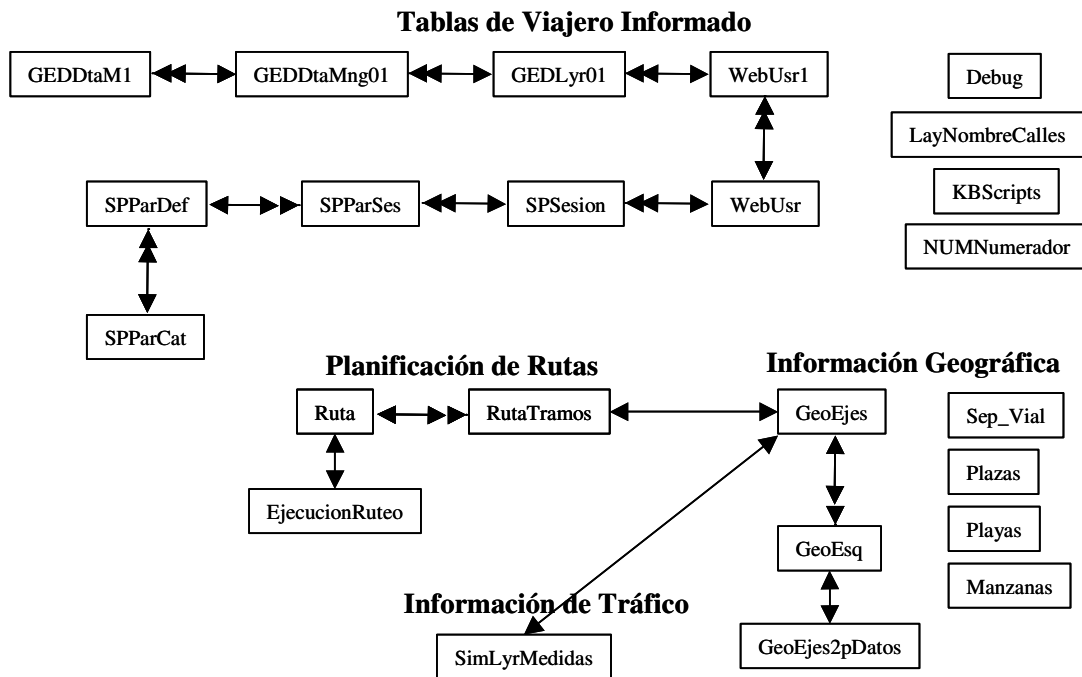


Figura 4-3. Diseño lógico de la Base de Datos.

La información sobre los atributos y las tablas se encuentra en el Apéndice A.

4.4.4 Consistencia de la Base de Datos.

Un aspecto que hay que considerar con respecto a la Base de Datos es la consistencia de la información sobre el nivel de congestión que hay en las calles. Esta información es generada por los sensores que se colocan en las calles que toman distintas medidas, por ejemplo la cantidad de vehículos que pasan y las velocidades que llevan. Si bien los sensores continuamente están recolectando información, solo la envían al Centro de Administración del Transporte cada cierto tiempo. Esto puede producir problemas cuando se utilice la información ya que se podría utilizar información desactualizada. Por otro lado hay que determinar qué representa la información que está en la Base de Datos. Se puede pensar que la Base de Datos contiene una “foto” del estado de la ciudad. Para que esa foto sea consistente es necesario que todos los datos sean válidos. Se llamará a un dato válido si este fue tomado en un período de tiempo en el cual refleja razonablemente las condiciones reales que representa. El sensor debería ser ajustado para que envíe nuevos datos cuando el período de validez de los datos anteriormente enviados, caducó.

En esta situación, lograr la consistencia de la Base de Datos, depende directamente de que los sensores envíen los datos en tiempo y forma adecuados. Si un sensor no envía datos o retrasa el envío por algún desperfecto, la Base de Datos quedaría inconsistente. Como es muy probable que los sensores fallen y la consistencia de la Base de Datos es un requisito importante para todas las aplicaciones que van a utilizar esos datos, hay dos opciones para manejar la consistencia: una opción es dejar que la consistencia no se de a nivel de toda la ciudad, sino a nivel de cada sensor y la otra opción es poner algún mecanismo que maneje el cambio de estado de toda la información de la ciudad a la vez y de ese modo, asegurar la consistencia global.

En la primer opción no se maneja un mecanismo que asegure tener una consistencia a nivel de toda la ciudad, sino a nivel de cada sensor. En este caso no se tiene en la Base de Datos una foto de un instante o un intervalo de tiempo dado, sino que se tiene la información que envían los sensores en forma independiente y que eventualmente no está coordinada con los demás sensores. En este caso cada aplicación debe encargarse de armar su propia foto global, si es que la necesita. La aplicación recuperará la información y evaluará si con los datos que tiene puede armar una imagen global del estado del tráfico que se ajusta a sus requerimientos. Por ejemplo un algoritmo de planificación de rutas puede considerar válido un dato que fue obtenido hace 15 minutos, pero si la planificación es para un vehículo de asistencia, como una ambulancia, entonces podría considerar inválidos todos los datos con más de 10 minutos y recurrir en su lugar a información histórica o solicitar al sensor que envíe la información. El inconveniente en este caso es que cada aplicación debe procurarse un mecanismo para armar la consistencia, aunque esto puede verse como una ventaja también, como es el ejemplo de la planificación de rutas de vehículos de emergencia que pueden manejar un nivel más ajustado en cuanto a la validez de los datos.

La otra opción implica que los datos que envían los sensores no ingresan directamente a la Base de Datos. Sino que son retenidos por alguna aplicación que cuando tiene la información completa de toda la ciudad los ingresa todos juntos a la Base de Datos. Si la información no es completa entonces puede recurrir a datos históricos, ingresar nuevamente el ultimo dato válido o utilizar algún simulador de tráfico para completar la información.

Otro problema a resolver ocurre luego de un cambio de estado, es decir alguno o todos los datos cambian y hay aplicaciones utilizándolos. Por ejemplo un algoritmo de planificación de ruta que mientras está computando una ruta la información sobre el nivel de congestión cambia. En este caso la aplicación debería de ser notificada del cambio de estado de modo de evaluar si debe desechar los cómputos realizados o en realidad puede continuar a pesar de que el estado cambió ya sea incorporando los datos del nuevo estado o continuar trabajando con el estado anterior. La aplicación puede evaluar, por ejemplo si en el nuevo estado no hay eventos suficientemente significativos que justifiquen desechar el trabajo realizado, por ejemplo si ocurriera un accidente que fuera en la zona por donde pasa la ruta planificada.

Si el algoritmo siempre opta por desechar el computo ya realizado y el cambio de estado ocurre con mucha frecuencia, puede ocurrir que el algoritmo nunca termine porque siempre está iniciando de nuevo la búsqueda de la ruta. Salvo en casos de incidentes, generalmente las condiciones del tráfico no cambian abruptamente en situaciones normales, por lo que podría permitirse que el algoritmo utilice los cómputos realizados en el estado anterior. Esto puede llegar a confundir al usuario, si es conciente de que existió un cambio en el estado y el algoritmo no desechó los cómputos realizados.

En la implementación realizada, se cambia el estado de toda la ciudad completa al mismo tiempo y los algoritmos de planificación incorporan estos nuevos datos, sin desechar los cómputos ya realizados.

4.5 Servidor de Mapas Digitales.

Antes de comenzar con en el tema del Servidor de Mapas Digitales se revisará el tratamiento de la información geográfica digital.

4.5.1 Información geográfica digital.

Existen dos formatos mediante los cuales se representa la información geográfica digitalizada. Uno es mediante imágenes digitalizadas, ya sean obtenidas desde satélites o escaneando mapas de papel, se le llama codificación raster [17]. El otro formato se conoce como codificación vectorial y codifica los elementos geográficos en términos de objetos geométricos básicos como puntos, líneas o polígonos.

La codificación raster no es la más adecuada para aplicaciones SAIV por que el espacio requerido para almacenar un mapa es mucho mayor y no resulta práctico en algoritmos en las cuales se deben utilizar modelos matemáticos. Sin embargo el formato vectorial es más adecuado en este caso, necesita menos espacio de almacenamiento, es más flexibles en la presentación y modela mejor las relaciones entre los elementos del mapa, ya que las estructuras de datos crean implícitamente relaciones [17]. En el resto del Capítulo solo se tratará el formato vectorial, que es el formato utilizado en este trabajo.

Los objetos geográficos tienen asociados *atributos* que almacenan información descriptiva, por ejemplo en el caso de una línea que representa una calle se almacenan el nombre, el largo, la dirección de circulación, etc.

Una *capa geográfica* es una colección de objetos geográficos que tienen el mismo significado, por ejemplo hay capas de líneas con información de calles o de polígonos

con información de las manzanas, etc. Entonces un *mapa digital* es una colección ordenada de capas geográficas eventualmente impresa en una imagen.

4.5.2 Representación de los elementos geográficos.

Las capas de información geográfica que se utilizaron fueron obtenidas del sitio web de la Intendencia Municipal de Montevideo [45], por lo tanto todos los datos se refieren a la ciudad de Montevideo. Los datos están en archivos con formato Shapefile. Este formato almacena la información geográfica en formato vectorial. Las capas que se utilizan son las siguientes:

- Ejes. Un eje de una calle, es una línea que pasa por el centro de la calle. Entre la información descriptiva está el nombre de la calle y el largo, entre otros atributos. Hay aproximadamente 23.000 tramos de ejes en la ciudad de Montevideo.
- Separadores viales. Son líneas que representan los cancheros que hay en el centro de algunas calles para separar las vías de circulación.
- Manzanas y plazas. Son polígonos que representan las manzanas y las plazas de la ciudad. Hay 7871 manzanas en la ciudad de Montevideo.
- Playas. Son líneas que representan las formas de las costas de Montevideo sobre el Río de la Plata.

Excepto por la capa de ejes, las demás capas solo se utilizan con fines visuales en la presentación del mapa.

La Intendencia Municipal de Montevideo tiene a su disposición, pero solo para su uso interno, más información relacionada con el sistema de transporte, por ejemplo la dirección de circulación en las calles o ubicación de semáforos, con la que hubiera sido interesante contar.

4.5.3 Información geográfica sobre las calles.

En esta Sección se describe la forma en que se utilizó la información de ejes para poder ser usada en los algoritmos de planificación.

Las calles normalmente están representadas de una manera estándar [17]. En particular la información de ejes de Montevideo sigue esas pautas. La representación de calles en mapas digitales vectoriales es la siguiente:

- Un *nodo* es un punto de intersección entre calles. Puede representar una esquina en la que convergen varias calles o también puede representar un extremo de una calle sin salida.
- Un *segmento* es una línea que forma una parte de una calle o camino que se encuentra entre dos nodos.
- Los *puntos de forma*, son colecciones ordenadas de puntos que sirven para modelar las porciones curvas de una calle. Los puntos de forma no incluyen los nodos. A la serie de segmentos que forman se les llama *segmentos de forma*.

Todos los puntos y segmentos se representan por sus coordenadas de latitud y longitud. Eventualmente podrían tener también altitud relativa, pero no es el caso. En la Figura 4-4 se puede ver a los nodos (puntos negros), los puntos forma (puntos rojos) y los segmentos en azul. La capa de Ejes está representada de esta manera, de hecho la Figura 4-4 corresponde a una sección de esa capa.

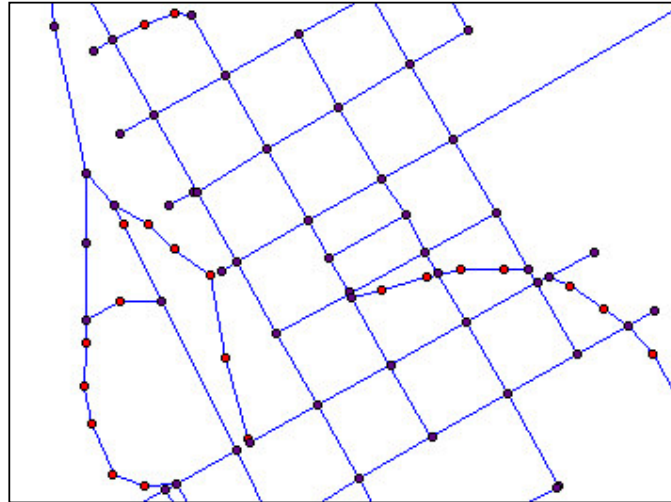


Figura 4-4. Elementos de calles y caminos en un mapa digital.

Los algoritmos de planificación de rutas y de ruteo de vehículos en general utilizan grafos para representar la topología de los caminos. Entonces lo que se hace para poder ejecutar los algoritmos es modelar las calles de la ciudad como un grafo. Para eso se toman como nodos del grafo los nodos en el mapa y como aristas del grafo se toman los segmentos. Los puntos y segmentos de forma no es necesario introducirlos en el grafo. Eso es porque la forma de la calle no es relevante para los algoritmos de planificación, se reduce la cantidad de nodos y aristas del grafo y como la distancia de los segmentos es un atributo en la capa, no es necesario la forma para calcular la distancia.

Si se dispone de información sobre el sentido de las calles entonces el grafo tendría aristas dirigidas, de acuerdo con la dirección del sentido de las calles. En este caso el grafo tiene aristas no dirigidas ya que no se dispone de esa información.

En la Figura 4-5 se puede observar gráficamente como se transforma el mapa con las calles en un grafo en el que se pueden ejecutar algoritmos de planificación, las líneas verdes son las aristas del grafo que se corresponden con las calles, conectando los nodos que son las esquinas.

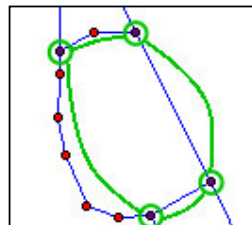


Figura 4-5. Formación del grafo a partir de los ejes.

Para mejorar el desempeño del algoritmo de carga en memoria del grafo, se preprocesó la capa de ejes antes de ingresar esa información en la Base de Datos, eliminando los puntos de forma. De este modo los algoritmos no necesitan hacer ese trabajo en cada ejecución.

El grafo está implementado con listas de nodos adyacentes que es la más adecuada cuando la cantidad de aristas por cada nodo es mucho menor que la cantidad de nodos [46]. Para optimizar la recuperación de nodos y aristas el grafo, que es una operación muy frecuente entre los algoritmos de planificación de rutas, se implementó un índice de nodos. El índice fue implementado utilizando una estructura de Hash [46] e indexa los nodos por sus coordenadas.

Existen dos versiones de la implementación del grafo sobre las calles, una para los algoritmos de planificación de rutas y otra para la simulación de tráfico. En la versión que se utiliza en los algoritmos de planificación, el grafo se va cargando desde la Base de Datos a medida que se necesita la información de otros nodos y aristas. Cuando el algoritmo examina un nodo del grafo, automáticamente se cargan los nodos adyacentes. De este modo no es necesario cargar el grafo con la ciudad completa, cuando se ejecuta el algoritmo. Se ahorra el tiempo de cargar todo el grafo y la memoria requerida para almacenarlo. En una primera implementación se cargaba toda la información de la ciudad en memoria, pero la demora en cargar el grafo completo es aproximadamente de 5 segundos, en un PC con 128 MB de memoria RAM, procesador Intel Pentium II y sistema operativo Windows 2000, que puede representar una demora considerable con respecto al tiempo total de ejecución. La forma utilizada de cargar los nodos, podría mejorarse si en lugar de cargar solo los nodos adyacentes, se cargara una sección más grande del grafo ya que se realizan menos consultas a la Base de Datos. Existen algunas estrategias para organizar la forma en que el grafo está almacenado en la Base de Datos, de modo que la recuperación a memoria necesite menos consultas en la Base de Datos [47].

La implementación del grafo que se utiliza para la simulación, carga una sección completa del mapa en el grafo. Luego la simulación trabaja sobre el grafo cargado, sin agregar nuevos nodos.

4.5.4 Servidor de Mapas.

Un Servidor de Mapas es un software que permite publicar e interactuar con mapas digitales desde páginas web. El servidor debe encargarse de leer la información de mapas, presentarlo en páginas web y permitir la interacción del usuario con el mapa. Los Servidores de Mapas son la evolución de los Sistemas de Información Geográfica hacia Internet.

El servidor de mapas seleccionado fue MapServer [48]. Es un servidor de código abierto desarrollado por la Universidad de Minnesota, en Estados Unidos. Las razones para seleccionarlo son:

- Funciona con bases de datos PostgreSQL extendidas con PostGIS. También funciona con archivos Shapefile.
- La instalación es extremadamente sencilla.
- El modo de operación se adapta al uso que se le quiere dar, en este proyecto.
- Es una herramienta de uso libre.

- La configuración y administración es sencilla.

El uso que se le pretende dar al servidor de mapas no es el uso para el que fue diseñado originalmente, pero puede adaptarse. El esquema simplificado de funcionamiento de MapServer se ve en la Figura 4-6.

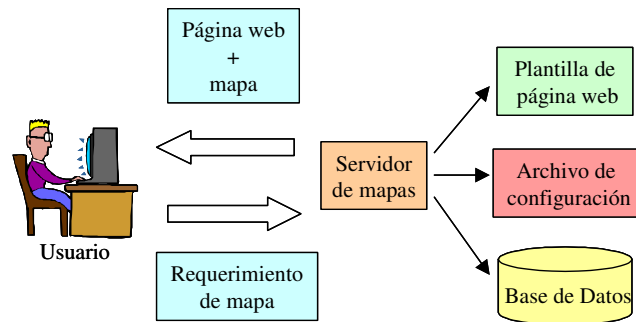


Figura 4-6. Funcionamiento del Servidor de Mapas MapServer.

El servidor recibe un requerimiento de un usuario, ya sea para presentar un mapa nuevo o realizar alguna acción sobre uno ya existente. Utilizando la información geográfica que se encuentra en la Base de Datos y un archivo de configuración que indica que capas debe presentar y de que modo, se genera un nuevo mapa. Luego utilizando una plantilla de página web, resuelve el requerimiento retornando una página web y la imagen del mapa. El problema es que el usuario está directamente interactuando con el servidor de mapas y si la página web tuviera que realizar algún tipo de procesamiento además de presentar un mapa, no le sería posible con este esquema. En nuevas versiones de MapServer se han desarrollado bibliotecas para algunos lenguajes de programación que permiten utilizar los servicios del servidor desde el programa que genera la página web y de ese modo resolver el problema. El modo de operación que se utilizó en este caso es el paradigma de Servicio Web (Web Services) [49]. Es decir que el servidor de mapas va a ser utilizado como si ofreciera un Servicio Web de creación de mapas. Este paradigma de operación es similar a los paradigmas cliente – servidor, es decir que un programa cliente utiliza servicios remotos proporcionados por otro programa, el servidor. En este caso la página web de Viajero Informado va a invocar al servidor de mapas para que le retorne mapas e información sobre ellos. Para poder llevar a cabo este mecanismo hay que determinar dos cosas, cual es el protocolo de comunicaciones por el cual se van a solicitar los servicios y se va a enviar la respuesta, y como será el pasaje de parámetros en la invocación y la respuesta. Para la comunicación se utiliza el protocolo HTTP (Hyper Text Transfer Protocol) [50]. La invocación al servidor se realiza ejecutando la URL (Uniform Resource Locator) [51] del servidor de mapas que también contiene los parámetros que se le envían. El resultado de la llamada esta en formato XML (Extensible Markup Language) [52]. En la Figura 4-7 se muestra el esquema de funcionamiento del sistema con el servidor de mapas:

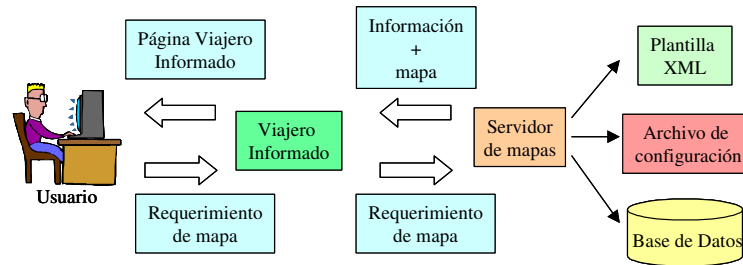


Figura 4-7. Invocación al Servidor de Mapas como Servicio Web.

En este caso el requerimiento del usuario es recibido por la página web de Viajero Informado y no directamente por el servidor de mapas. La página web invoca al servidor de mapas y recibe la respuesta que contiene dos datos:

- La URL a donde el servidor de mapas creó la imagen con el mapa, para que la página web pueda mostrarlo.
- Información del estado de la vista del mapa para poder reproducirla en la próxima invocación. En este caso es suficiente con que el servidor envíe la *extensión* del mapa. La extensión son las coordenadas del borde superior izquierdo e inferiores del derecho del mapa. Con esta información el servidor sabe exactamente cual es la zona que el usuario esta viendo en el mapa.

Estos datos son almacenados en la Base de Datos junto con otra información sobre el mapa, por ejemplo las capas que el usuario esta viendo. Esta información de sesión es administrada por Viajero Informado, ya que en la etapa en que se desarrolló esta parte del prototipo, MapServer tenía pocas facilidades para el manejo de datos de sesión. En las versiones más nuevas como la 3.6 tienen mejores facilidades.

Con respecto al manejo de la sesión del usuario los servidores de mapas tienen dos posibilidades, una es que la maneje la aplicación que está utilizando el servicio y la otra es que el propio servidor de mapas la maneje automáticamente. En este caso fue necesario desarrollar el manejo de sesiones. Esta tarea no es sencilla porque implica procesar la información de la sesión para poder pasarle los parámetros al servidor de mapas utilizando la URL. Dependiendo de la acción que esta ejecutando el usuario, la forma de invocación al servicio puede cambiar radicalmente. Por lo tanto es necesario crear un motor de generación de invocaciones al servidor para que la interacción con el usuario sea implementada. La integración con el servidor de mapas se explica con más detalle en la Sección 4.7.

4.5.5 Interacción con el mapa.

Una de las formas más importantes de interactuar con los mapas digitales es realizando operaciones directamente sobre la imagen. Las operaciones más frecuentes son zoom, paneo o identificación. El zoom consiste en acercar o alejar la vista de la zona seleccionada. El paneo es movilizar la vista del mapa sin alejarla o acercarla. La identificación consiste en seleccionar algún elemento del mapa y ver los atributos asociados. MapServer ofrece facilidades para soportar estas operaciones y algunas más.

El problema está en la interacción que ofrece el HTML (Hypertext Markup Language) [53] con las imágenes. Hay dos modos con los cuales se puede interactuar con las imágenes en HTML. Uno, es simplemente asociando una URL a una imagen. Cuando se hace clic sobre la imagen se ejecuta la URL asociada y se pasa como parámetros las coordenadas del clic en la imagen. El problema de este modo de operar es que solo ofrece la posibilidad de hacer un clic en el mapa, no se puede marcar un rectángulo o algún polígono más complejo en la selección. Además el pasaje de las coordenadas como parámetros no es compatible con el modo de operación de MapServer. El otro modo de operación es indicar al HTML que la imagen sea interpretada como un mapa. En este caso se indican zonas en el mapa, donde cada una es asociada a una URL distinta. Este modo de operación es muy útil para implementar acciones como la identificación, pero para el resto de las operaciones no es útil.

Para resolver este problema los Servidores de Mapas utilizan dos alternativas distintas, una es dejar que el mapa sea una imagen y mejorar la interactividad desde la página web utilizando JavaScript [54] o Applets Java [55]. La otra opción es que el mapa no sea realmente una imagen sino que es un Applet Java embebido en la página web y las acciones sobre el mapa las ejecuta el Applet conectándose directamente con el Servidor de Mapas. En este caso la página web no se somete al Servidor Web, sino que el Applet se encarga de conectarse al Servidor de Mapas y requerir el nuevo mapa.

En Viajero Informado se seleccionó dejar que el mapa sea realmente una imagen y extender la interactividad incluyendo un Applet Java en la página web. El Applet seleccionado se llama jBox, es de uso libre y fue desarrollado por Greenwood Mapping Inc. [56]. El funcionamiento del mapa es el siguiente, cuando el usuario hace un clic o un rectángulo sobre el mapa, el Applet carga los valores de las coordenadas en variables y envía el requerimiento al servidor web (somete la página web). La página web se encarga de enviar al Servidor de Mapas el requerimiento para que retorne un nuevo mapa y de presentárselo al usuario.

4.5.6 Configuración del Servidor de Mapas.

Los componentes que se deben configurar en una aplicación que utilice MapServer son: archivo de configuración del mapa (map file) y el archivo de resultados (template file) [48].

El archivo de configuración del mapa contiene todos los detalles sobre la configuración de las capas geográficas que se utilizan e información de configuración como ser el acceso a los datos o donde colocar las imágenes de los mapas creados. En Viajero Informado se utilizan cuatro archivos de configuración:

- querymap.map. Esta configuración sirve para presentar mapas con las rutas que fueron planificadas utilizando información geográfica.
- sesionmap.map. Esta configuración es utilizada para el mapa principal que aparece en la pantalla inicial de Viajero Informado y que no muestra información de rutas planificadas o niveles de congestión de vehículos.

- simulationquerymap.map. Esta configuración presenta las rutas planificadas utilizando la información de congestión y además presenta en el mapa los distintos niveles de congestión de las calles.
- simulationmap.map. Con esta configuración se puede ver la información de congestión de las calles de la ciudad.

Para retornar los resultados se utiliza el archivo de resultados llamado “sesionmap.html” que tiene el siguiente contenido:

```
<xml>
<image> [img] </image>
<mapext> [mapext] </mapext>
</xml>
```

Cuando MapServer recibe un requerimiento de un mapa utiliza este archivo para retornar los resultados. MapServer lee este archivo y reemplaza las variables predefinidas (que están entre corchetes) por los valores correspondientes. En este caso [img] es reemplazado por la URL de la imagen con el mapa y [mapext] es reemplazada por la extensión del mapa. MapServer maneja distintos tipos de variables predefinidas, en este caso solo se utilizaron estas dos, pero existen varias más. Luego que realizó las sustituciones correspondientes, retorna el archivo. La idea original de los creadores de MapServer es utilizar a los archivos de resultados como páginas HTML para que el usuario interactúe directamente con el servidor, pero si en lugar de código HTML se pone código XML, se permite que el servidor pueda interactuar como un Servicio Web, como en este caso.

4.5.6.1 Invocación al Servidor de Mapas.

El servidor de Mapas es invocado utilizando una URL en donde se le pasan distintos parámetros que indican cuales son las acciones que debe ejecutar. Se verá como ejemplo de invocación al servidor de mapas, las dos formas que se utilizan de invocación, una para retornar el mapa de la ciudad y la otra para desplegar planificaciones de rutas.

Mapas de la ciudad.

La URL que se utiliza es similar a la siguiente:

```
http://localhost/cgi-bin/mapserv36/mapserv.exe?
map=mapserv36\mapa\sesionmap.map&mode=browse&zoom=0&
layer=manzanas&layer=playas&layer=plazas&layer=sep_vial
```


La explicación de la URL es la siguiente:

URL	Detalle
http://localhost/cgi-bin/mapserv36/mapserv.exe?	Localización en el servidor Web del servidor de mapas.
map=mapserv36\mapa\sesionmap.map	Localización del archivo map con la especificación del mapa a presentar.
mode=browse	Modo en que está trabajando con el mapa. En este caso browse indica que el usuario está interactuando con el mapa.
zoom=0	Indica si es necesario aplicar la operación de zoom al mapa. En este caso no se hace zoom.
layer=manzanas&layer=playas&layer=plazas&layer=sep_vial	Indica las capas de información que se van a activar en mapa, es decir las capas que se van a ver.

Cuando el usuario realiza alguna operación sobre el mapa, por ejemplo centra el mapa o realiza un zoom, el valor de los parámetros cambian y eventualmente aparecen nuevos parámetros [48]. El sitio web tiene desarrollado un mecanismo para generar distintas URL dependiendo de las acciones que quiera hacer el usuario.

Mapas con rutas.

La diferencia en este caso es que para presentar una ruta es necesario hacer una consulta utilizando el identificador de la ruta que se quiere presentar. La URL en este caso es:

*http://localhost/cgi-bin/mapserv36/mapserv.exe?
map=mapserv36\mapa\simulationquerymap.map&qlayer=ruteo&rutaid_par=184&
mode=itemnquery&mapext=shapes&layer=simulation&layer=manzanas&layer=pla
yas&layer=plazas&layer=sep_vial*

La explicación de la URL es la siguiente:

URL	Detalle
http://localhost/cgi-bin/mapserv36/mapserv.exe?	Localización en el servidor Web del servidor de mapas.
map=mapserv36\mapa\simulationquerymap.map	Localización del archivo map con la especificación del mapa a presentar.
mode=itemnquery	Modo en que está trabajando con el mapa. En este caso itemquery indica que el usuario realizo una consulta en el mapa.
qlayer=ruteo	Este parámetro indica cual es la capa que esta siendo consultada, en este caso la capa se llama: ruteo.
ruta_id_par=184	Parámetro para hacer la consulta. Lo que esta indicando este parámetro es que se quiere la ruta que se encuentra almacenada en la Base de Datos, cuyo identificador es el 184.
mapext=shapes	Indica que la extensión del mapa que se muestra abarca a todos los resultados de la capa consultada.
layer=simulation&layer=manzanas&layer=playas&layer=plazas&layer=sep_vial	Indica las capas de información que se van a activar en mapa.

4.6 Planificación de la ruta.

La planificación de la ruta consiste en determinar cual será el mejor camino por el cual el conductor llegará al destino. El concepto de “mejor” depende del criterio que se considere para optimizar. Un camino puede ser el mejor porque es el que recorre la menor distancia, sin embargo el camino por el cual se llegue en menor tiempo al destino puede ser otro distinto. Dependiendo del criterio que se quiere tener en cuenta, es la información que deben utilizar los algoritmos. Para planificar el camino más corto en términos de distancia, solo se utiliza la información del mapa de las calles. Pero si se quiere la ruta con la que se llegue en menor tiempo, se utiliza información sobre los niveles de congestión de las calles.

Los métodos de planificación de rutas podrían clasificarse en dos tipos, los que planifican las rutas para múltiples vehículos simultáneamente y los que planifican las rutas para cada vehículo individualmente. Desde el punto de vista de un sistema de navegación que utiliza un usuario, solo tiene sentido utilizar un algoritmo que planifique la ruta de ese usuario, ya que él no tendrá acceso a los requerimientos de planificación de otros usuarios.

Existe un problema al utilizar este enfoque: ¿qué ocurriría si un algoritmo diera a todos los usuarios la misma ruta? Esto podría derivar en que todos esos usuarios que recibieron esa planificación “repetida” congestionen rápidamente la zona planificada. Para evitar este problema la solución inmediata es hacer que el algoritmo pueda considerar las planificaciones que ya ha presentado a los usuarios, de modo de ir anticipando el nivel de congestión que él mismo genera. Sin embargo, los experimentos llevados a cabo por Esser y Schreckenberg [57] indican que no es necesario utilizar ese tipo de algoritmos. Si el tiempo de actualización de la información es suficientemente corto y se actualiza la ruta mientras se conduce, no necesariamente se producen congestiones debido al algoritmo, incluso si hay muchas personas utilizando el sistema en relación a las personas que no lo utilizan. Este resultado puede explicarse intuitivamente de la siguiente manera. A medida que los vehículos tomen la misma ruta inducidos por el algoritmo, los sensores en las calles registrarán el incremento en la congestión. Esa información será introducida en las bases de datos de los SAIV de modo que si existiesen nuevos requerimientos de rutas, esa congestión ya será tomada en cuenta por el algoritmo.

El problema de la planificación de rutas es conocido en la literatura como encontrar el camino más corto desde un nodo a otro en un grafo. Este problema es ampliamente estudiado y existe una serie de algoritmos y técnicas para resolverlo, en particular el algoritmo de Dijkstra [17] es uno de los más populares. Sin embargo Viajero Informado utiliza otro algoritmo llamado A* [17], que es más conveniente para resolver este problema.

4.6.1 Modelado del problema.

En los algoritmos de ruteo de vehículos o de encontrar el camino más corto desde un origen a un destino, se utilizan grafos para representar la topología de la ciudad. En

esta Sección se profundiza sobre el significado del costo que se le asocia a las aristas del grafo. Cuando se utiliza un algoritmo de búsqueda genérico como A* es necesario definir dos aspectos del problema para que se adapte al caso particular. Un aspecto es definir cual va a ser el grafo, que en este caso se carga dinámicamente a partir de las calles. El otro aspecto, es definir el tipo de costo que se va a utilizar. El algoritmo de planificación utiliza los costos para determinar el camino menos costoso hacia el destino. El significado que tenga el costo determinará el significado que tiene el camino encontrado, por ejemplo si el costo corresponde a la distancia entre las esquinas, entonces el camino encontrado será el que recorra menos distancia, si el costo es el tiempo que se necesita para atravesar la calle, entonces el camino será el que permite llegar más rápido al destino.

Para el correcto funcionamiento de los algoritmos, los costos no pueden ser negativos, sino el algoritmo tendería a incorporar infinitamente esas aristas. El costo en la implementación realizada, es un valor numérico real.

Las funciones que determinen el costo de una arista pueden llegar a ser complejas, por ejemplo si existiese información sobre semáforos, cantidad de carriles de las calles o si el usuario prefiriera no viajar por determinadas calles, entonces la función debería ponderar todas estas condiciones extras para lograr un único valor numérico de costo.

Por otro lado, hay costos que es muy difícil cuantificar, pero deberían ser tenidos en cuenta en la planificación. Por ejemplo, seguridad, nivel de conocimiento del usuario de la ruta o preferencias del usuario con respecto a los barrios que debe transitar.

La calidad de una planificación depende mucho de la cantidad de información que se tiene en cuenta al generarla e incluirla como costo. Algunos de los factores que determinarán la calidad de la ruta son:

- Distancia recorrida.
- Tiempo del viaje.
- Velocidad de viaje.
- Cantidad de giros.
- Semáforos que hay que atravesar.
- Nivel de congestión de las calles.
- Seguridad, en cuanto a los barrios que se atraviesen o la frecuencia de accidentes que hay en determinados cruces de calles, etc.

En este trabajo se tienen en cuenta por un lado la distancia recorrida y por otro el tiempo del viaje. Para eso se implementó un algoritmo que opcionalmente utiliza como costo la distancia recorrida o el tiempo de viaje, dependiendo de lo que el usuario quiera utilizar. Es decir que el algoritmo de planificación de rutas maneja dos criterios de búsqueda diferentes. En las siguientes secciones se argumenta porque no fue utilizado el algoritmo de Dijkstra y se describe el algoritmo A*.

4.6.2 Algoritmo de Dijkstra.

Uno de los algoritmos más populares para resolver el problema de encontrar caminos más cortos es el algoritmo de Dijkstra [17]. El algoritmo utiliza una técnica golosa (greedy) para encontrar el camino más corto desde un nodo de origen a todos los demás nodos del grafo. Para resolver este problema en particular, como solo interesa el camino más corto al nodo destino, cuando se encuentra el camino más corto, se detiene. A este algoritmo se le conoce como algoritmo de Dijkstra modificado [17]. El algoritmo de Dijkstra presenta algunas desventajas que hacen que no sea el algoritmo más adecuado para resolver el problema de la planificación de rutas en ciudades:

- Características del grafo. Los grafos que modelan las ciudades tienen un tamaño muy grande con miles de nodos y aristas. El algoritmo de Dijkstra explora (recupera la información del nodo y de los nodos y aristas adyacentes) un conjunto muy grande de nodos y aristas antes de encontrar el nodo de destino. Si tenemos en cuenta que el grafo es grande, entonces el algoritmo realizará demasiado trabajo extra en encontrar la solución.
- Características del algoritmo. El algoritmo va encontrando los caminos más cortos desde el origen a todos los nodos y se detiene cuando encuentra el destino. Es decir que no trata de dirigir la búsqueda hacia el nodo de destino. Esto determina que para encontrar un camino, el algoritmo explora nodos que no van a ser incluidos en la solución y que no están en la dirección del nodo de destino.

Si bien se han propuesto distintas variaciones del algoritmo para hacer más eficiente la búsqueda del algoritmo de Dijkstra, los problemas mencionados persisten [17]. Por ejemplo, la búsqueda bidireccional, que ejecuta dos instancias paralelas del algoritmo, una desde el origen al destino y otra desde el destino hacia el origen. Cuando se encuentran dos nodos ya explorados por las instancias entonces se construye la solución.

Afortunadamente existe un algoritmo llamado A^* que utilizando información heurística, es capaz de encontrar una ruta de la misma calidad que el algoritmo de Dijkstra, pero explorando menos nodos.

4.6.3 Algoritmo A^* .

A^* es un algoritmo de búsqueda respaldado por información que pertenece al método de búsqueda preferente por lo mejor (best first search). Es muy popular en la planificación de rutas y ha sido exitosamente utilizado en otras áreas, como Inteligencia Artificial y Robótica, ver [17] y [58].

A^* es un algoritmo genérico de búsqueda que utiliza información heurística para determinar cual es el mejor camino hacia el destino. La información que utiliza es el costo estimado desde el nodo explorado hacia el nodo destino. Esta información es proporcionada por una función llamada *función heurística* $h'(n)$. Si n es el nodo destino, la función debe cumplir que $h'(n) = 0$.

El costo de la ruta estará dado por la suma de los siguientes valores:

- $g(n)$ – indica cuanto es el costo de la ruta desde el nodo de origen hacia el nodo n .
- $h'(n)$ – indica cuanto es el costo estimado de la ruta desde el nodo n hacia el nodo destino.

La función del costo es entonces:

$$f'(n) = g(n) + h'(n)$$

donde $f'(n)$ es el costo estimado de la solución, de menor costo, pasando por el nodo n .

La función $f'(n)$ también es un función heurística ya que utiliza el resultado de $h'(n)$. Para encontrar la solución de menor costo A^* , ira probando primero con el nodo cuyo valor de f' es el menor. Si h' presenta ciertas condiciones, se puede demostrar [58] que esta estrategia es completa (garantiza encontrar una solución, si existe) y óptima (se encuentra la mejor solución si existiesen varias). La condición que se le debe imponer a h' es que siempre subestime el costo que implica alcanzar el nodo destino. Es decir que el costo real debe ser siempre mayor o igual al costo estimado por h' . A las funciones que cumplen con esta propiedad se las conoce como *heurísticas admisibles*. Generalmente, las heurísticas admisibles presentan monotonicidad. Esto quiere decir que la función cumple la desigualdad triangular [58] y por lo tanto el valor de f' no decrece a medida que la búsqueda se hace más profunda. Entonces, como f' nunca disminuye y el algoritmo expande el nodo que tiene el costo f' más bajo, se van añadiendo nodos a la búsqueda en bandas concéntricas en las que el valor del costo f' va aumentando.

En algoritmos como Dijkstra en el que el costo es uniforme ($h' = 0$), estas bandas se generan en forma circular en torno al nodo de partida. En el caso de A^* las bandas tienden a tener forma ovoide y dependiendo de la función heurística h' tienden a estar más apretadas en torno a la ruta óptima. La Figura 4-8 muestra esta situación.

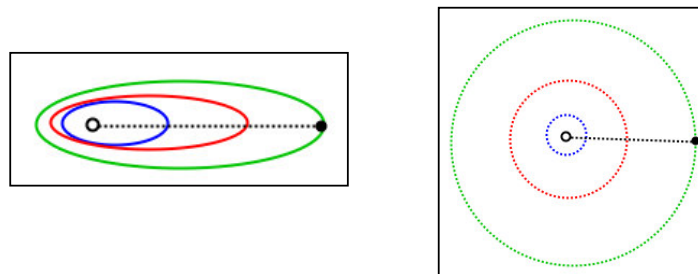


Figura 4-8. Forma de los espacios de búsqueda de A^* y Dijkstra respectivamente.

Analizado el tiempo de ejecución de A^* se puede determinar que tiene igual desempeño que el algoritmo de Dijkstra, si se utiliza notación O . En [17] se menciona un análisis del tiempo de ejecución que explica la diferencia en el desempeño entre ambos. Sea b el factor de bifurcación, que es el promedio de la cantidad de aristas que salen de un nodo a otro en el grafo y sea d la profundidad del camino más corto de la ruta solución del nodo origen al destino. Entonces el tiempo de ejecución (de ambos algoritmos) es $O(b^d)$, pero por la información heurística utilizada el factor de

bifurcación b que maneja en la práctica A^* es menor que el que maneja Dijkstra. Del mismo modo que ocurre en el algoritmo de Dijkstra, si se utilizan estructuras de datos que realicen operaciones de búsqueda eficientemente, como Heaps, el tiempo de ejecución disminuye [17].

4.6.3.1 Algoritmo.

A continuación se presenta un pseudocódigo de la implementación utilizada de A^* [17]:

```

1. ABIERTA = {(no, 0, 0, 0)};
2. CERRADA = {};
3. mejor = 0;
4. While (mejor <> nd )
5. begin
6.   mejor = NodoMenorCosto(ABIERTA);
7.   Remover(ABIERTA, mejor);
8.   Insertar(CERRADA, mejor);
9.   if (mejor <> nd ) then
10.  begin
11.    foreach n = NodoAdyacente(mejor, G)
12.    begin
13.      costog = g(mejor) + costo(mejor, n);
14.      if Pertenece(n, ABIERTA) then
15.      begin
16.        if ObtenerCostoG(n,ABIERTA) < costog then
17.        begin
18.          ActualizarDatos(ABIERTA,n,costog, mejor);
19.        end
20.      end
21.      else
22.      begin
23.        if Pertenece(n, CERRADA) then
24.        begin
25.          if (ObtenerCostoG(n, CERRADA)< costog) then
26.          begin
27.            Eliminar(n, CERRADA);
28.            Insertar(ABIERTA,n,costog,mejor);
29.          end
30.          end
31.          else
32.          begin
33.            costof' = costog + h(n);
34.            Insertar(ABIERTA,n,costof',mejor);
35.          end
36.        end
37.      end
38.    end
39.  end
40. r = RecuperarRuta(nd, ABIERTA, CERRADA);

```

Variables:

no (numérico) – Nodo de origen.

nd (numérico) – Nodo de destino.

mejor (numérico) – Identificador del próximo nodo a explorar.

n (numérico) – Identificador de nodo.

ABIERTA, CERRADA – Listas que almacenan estructuras con el siguiente formato: (Identificador de nodo, $g(n)$, $f'(n)$, puntero de retroceso).

G – grafo con los caminos.

costog (numérico) – se almacena temporalmente el valor de $g(n)$.

costof' (numérico) – se almacena temporalmente el valor de $f'(n)$.

Funciones:

NodoMenorCosto(Lista): nodo. Esta operación retorna el nodo de la lista que tenga menor valor de $f'(n)$.

Remover(Lista, nodo). Elimina un elemento de la lista.

Insertar(Lista, nodo). Inserta un elemento en la lista.

NodoAdyacente(nodo, Grafo):nodo. Retorna los nodos adyacentes en el grafo G a un nodo dado.

Costo(nodo1, nodo2):numérico. Retorna el costo desde el nodo1 al nodo2.

ObtenerCostoG(nodo, Lista):numérico. Retorna el costo ($g(nodo)$) del nodo que está en la lista.

ActualizarDatos(Lista, nodo, costog, nodopadre). Actualiza la información asociada al nodo en la lista.

Pertenece(n, CERRADA):bool. Comprueba si el nodo pertenece a la lista.

La implementación utiliza dos listas llamadas ABIERTA y CERRADA para almacenar los nodos que aún no fueron explorados en ABIERTA y los que ya fueron explorados en CERRADA. Estas listas además de almacenar los nodos también almacenan información sobre las funciones de costo g , f' e información sobre el nodo anteriormente explorado llamada puntero de retroceso, que indica el nodo padre en el árbol de búsqueda. Con la información de costos, es que se decide el orden de los nodos a explorar. La información del nodo anteriormente explorado sirve para cuando es encontrado el nodo de destino, poder recuperar el camino realizado, recorriendo en forma inversa el árbol de búsqueda.

En las primeras tres líneas iniciales se realiza la inicialización de las variables. En la línea 6 se recupera el nodo que tiene menor valor de f' en la lista ABIERTA y se lo coloca en la lista CERRADA. Luego por cada nodo adyacente en el grafo se realizan distintas operaciones dependiendo de sí el nodo ya fue explorado o no.

Si el nodo pertenece a la lista ABIERTA, se le actualiza el costo en el caso de que se haya encontrado un camino más corto para llegar a él. Si el nodo se encuentra en la lista CERRADA y se actualiza el costo (línea 27), entonces se coloca en la lista ABIERTA para que sea explorado nuevamente. Si el nodo no se encuentra en ninguna de estas listas (línea 33), se calcula f' del nodo y se inserta en ABIERTA.

Si se recupera el nodo destino entonces el algoritmo termina y calcula la ruta desde el nodo origen al destino (línea 40).

Existen variaciones de la implementación de A* que utilizan una cantidad de memoria limitada al ejecutar. Estas implementaciones son muy interesantes para las

planificaciones de rutas, ya que en ciudades muy grandes la cantidad de memoria que utiliza el algoritmo en una ruta larga puede resultar muy importante [58].

4.6.3.2 Funciones heurísticas.

Se utilizaron dos funciones heurísticas distintas en la implementación, una considera la ubicación geográfica del destino mientras que la otra utiliza el tiempo del viaje. No fue necesario implementar criterios de podas adicionales a los que se proponen como funciones heurísticas.

4.6.3.2.1 Ubicación geográfica.

Esta función evalúa el costo utilizando la distancia Euclídea que hay entre el nodo explorado y el nodo destino. Es decir que en este caso el costo en el problema está dado por la distancia de la ruta. Como la distancia Euclídea cumple la desigualdad triangular y siempre la distancia Euclídea es una subestimación de la distancia de una calle (por ejemplo si la calle tiene una curva, entonces la distancia Euclídea, será menor que la distancia que se debe recorrer para llegar de una esquina a otra), esta función h' es una heurística admisible [17].

$$h'(n) = \text{distanciaEuclídea}(n, \text{destino})$$

En la Figura 4-9 puede verse gráficamente el funcionamiento, el nodo origen (azul), el destino (rojo), un nodo intermedio (negro) y la función $h'(n)$.

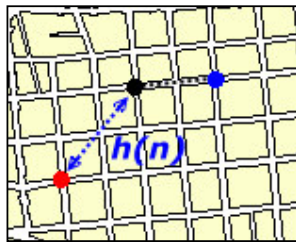


Figura 4-9. La función utiliza la localización del destino.

4.6.3.2.2 Tiempo del viaje.

En este caso el costo del problema es el tiempo en que se demora en llegar al destino, que depende de las condiciones de la red de calles. La función heurística en este caso es la siguiente:

$$h'(n) = \text{distanciaEuclídea}(n, \text{destino}) / V_{\text{MAX}}$$

Es decir que la función computa la distancia Euclídea entre el nodo explorado y el destino y lo divide entre la velocidad máxima permitida en la ciudad. Intuitivamente estima como si se pudiera llegar al destino por un camino directo y conduciendo a

velocidad máxima. La admisibilidad de la función nuevamente queda asegurada ya que la distancia Euclídea entre dos esquinas, en una ciudad es menor o igual que la distancia efectiva que hay que recorrer para llegar de una a otra. Por otro lado la velocidad real a la que se puede conducir será menor o igual a la velocidad máxima permitida por la ley.

4.6.4 Implementación.

Los algoritmos fueron implementados en el lenguaje C++ utilizando Visual Studio de Microsoft. El acceso a la Base de Datos se realiza mediante ODBC [59]. Para facilitar la implementación del acceso a la Base de Datos se utilizó un biblioteca de clases de código abierto desarrollada por MPH [60]. También se utilizaron otras clases de esa biblioteca como un generador de números aleatorios y clases para dejar bitácoras de la ejecución. Para el manejo de parámetros en archivos de inicio se utilizó una clase de código abierto [61]. Para el manejo de la información en archivos Shapefile, se utilizó la biblioteca ShapeLib [62]. El proyecto tiene aproximadamente 40 clases y 6.000 líneas de código. Fue utilizado para el desarrollo y la ejecución un PC con 128 MB de memoria RAM, procesador Intel Pentium II, disco duro de 40 Gigabytes y sistema operativo Windows 2000.

4.6.4.1 Diseño de clases.

En la Figura 4-10 se ve un Diagrama de Clases del módulo encargado de realizar la planificación de las rutas [63]. En los cuadrados se pueden ver los nombres de las distintas clases C++ y las flechas indican dependencias, es decir que una clase utiliza funcionalidades de otra. Existen algunas clases más pero se omitieron en el diagrama para facilitar la comprensión. Estas clases en su mayoría pertenecen a distintas bibliotecas utilizadas como ShapeLib o clases para el acceso a la Base de Datos.

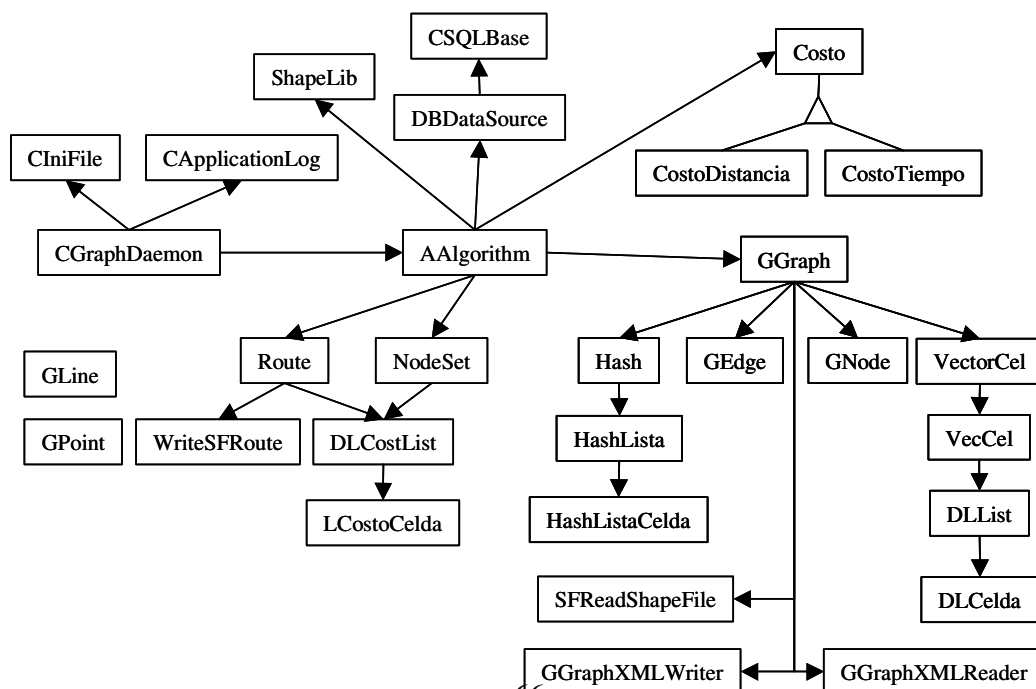


Figura 4-10. Diagrama de clases de la aplicación de Planificación de Rutas.

Descripción de las clases:

GGraph – Implementa un grafo con las calles de la ciudad. El grafo es implementado con listas de nodos adyacentes, ya que esta implementación es más eficiente en grafos dispersos (con pocas aristas por nodo) como los de calles.

GEdge – Aristas del grafo.

GNode – Nodos del grafo.

Hash, HashLista y HashListaCelda – Implementan un índice sobre los nodos del grafo. De este se puede recuperar eficientemente un nodo del grafo, dando sus coordenadas x e y. El índice es implementado utilizando un Hash abierto [46].

VectorCel, VecCel – Implementan un vector automáticamente redimensionable de listas DList (aristas) y un nodo.

DList, DLCelda – Implementan un lista de aristas.

SFReadShapefile – Carga desde un archivo Shapefile las calles de la ciudad. Esta forma de cargar las calles no es utilizada por que demora demasiado tiempo en cargar la información en memoria.

GGraphXMLWriter – Permite salvar las calles de la ciudad como un archivo XML. En la implementación actual no es utilizada porque este método para manejar la información demora demasiado tiempo en cargarse en memoria.

GGraphXMLReader – Esta clase complementa a la clase anterior, permitiendo leer la información de calles desde un archivo XML. Tampoco es utilizada en la implementación actual.

Costo, CostoDistancia, CostoTiempo – Implementan el tratamiento de los distintos tipos de costo posibles en una planificación de una ruta, ya sea cuando se utiliza la distancia o el tiempo en atravesar una calle.

ShapeLib – Conjunto de clases que permiten operar con archivos Shapefile. Se utilizan para salvar en un archivo Shapefile los conjuntos de nodos utilizados durante la ejecución del algoritmo de planificación.

AAlgorithm – Implementación del algoritmo A*.

DLCostList, LCostoCelda - Implementan listas de nodos y costos.

Route – Ruta planificada.

WriteSFRoute – Salva una ruta en un archivo Shapefile.

NodeSet – Implementa las listas utilizadas por el algoritmo A* para almacenar los nodos que está explorando.

DBDataSource – Esta clase se utiliza para implementar todos los accesos a la Base de Datos. Aunque no esta indicado en la Figura 4-10, para preservar la claridad del diagrama, el grafo la utiliza para cargar la información sobre la ciudad y el algoritmo de planificación de rutas la utiliza para salvar la ruta.

CSQLBase – Esta clase, junto con otras que se omitieron en el diagrama para mantener la claridad, sirven para facilitar el acceso a la Base de Datos utilizando la tecnología ODBC [60].

GLine, GPoint – Implementan una línea y un punto en dos dimensiones. Son utilizadas por varias clases para realizar distintas operaciones geométricas.

CIniFile – Permite acceder parámetros configurados en archivos de inicio.

CApplicationLog – Permite escribir en una bitácora la traza de la ejecución del programa.

CGraphDaemon – Invocación al algoritmo de planificación de rutas.

4.6.4.2 Parámetros configurables.

Algunos parámetros utilizados por el algoritmo de planificación de rutas son configurados desde un archivo de inicio externo al programa. A continuación se describen estos parámetros.

Parámetro	Descripción
Log	1 o 0 según se quiera la traza de la ejecución del programa.
Logpath	Ubicación del archivo de la traza.
Dsn	Nombre del Data Source para el acceso a la Base de Datos
Savetoshpfiles	1 o 0 según se quiere que la ruta planificada sea salvada a un archivo Shapefile
routeshp	Ubicación del archivo .shp de la ruta.
Routedbf	Ubicación del archivo .dbf de la ruta.
Saveusedpoint	1 o 0 según se quiere que los nodos utilizados por A* sean salvados en un archivo Shapefile. Se salvan tres conjuntos de puntos, los conjuntos Abierto, Cerrado y otro con todos los puntos utilizados.
routeusedp	Ubicación de los archivos Shapefile con los puntos usados.
routeopenshp	Ubicación de los archivos Shapefile con el conjunto de puntos Abierto.
Routecloseshp	Ubicación de los archivos Shapefile con el conjunto de puntos Cerrado.
maxspeed	Velocidad máxima (en celdas) a la que pueden ir los vehículos.
celllengt	Largo de la celda.

En la Figura 4-11 se lista el archivo de configuración utilizado actualmente.

```
# Archivo de configuración de parámetros

[Log]
log=0
logpath=C:\progs\log.txt

[DataBase]
dsn=atispq

[RoutingAlgorithm]
savetoshpfiles=0
routeshp=C:\progs\shp\route.shp
routedbf=C:\progs\shp\route.dbf
saveusedpoint=0
routeusedp=E:\desa\archivos\aupoints
routeopenshp=E:\desa\archivos\open
routecloseshp=E:\desa\archivos\close

[SimulatorVehicleConfig]
maxspeed=4

[SimulatorCells]
celllengt=5
```

Figura 4-11. Archivo de configuración.

4.7 Sitio Web.

El sitio web, Viajero Informado, es un conjunto de páginas web que permiten que el usuario interactúe con la información y las herramientas. Tiene dos componentes: las páginas web que forman el sitio de tráfico y las herramientas de administración del sitio, que también son páginas web. Para desarrollarlo se utilizó Genexus [44] y el lenguaje de generación seleccionado fue Java.

En la siguiente Sección se describe la funcionalidad del sitio web y luego se describe la implementación.

4.7.1 Viajero Informado.

La Figura 4-12 presenta la página principal de Viajero Informado. En la parte central se puede ver el mapa del sistema. Al costado derecho del mapa están las herramientas para trabajar con el mapa. En el costado izquierdo del mapa están las opciones para que el usuario se registre y navegue por el sitio.



Figura 4-12. Página principal de Viajero Informado.

4.7.1.1 Herramientas de Acción y Leyenda.

Estas herramientas permiten manejar la vista del mapa. Las herramientas de Acción presentan las opciones de Acercar, Centrar y Alejar. Si el usuario hace clic en la imagen del mapa, la acción que se ejecutará será la que se seleccionó en Acción. El usuario también puede trazar un rectángulo sobre el mapa y en ese caso la imagen se centra en el rectángulo trazado. La herramienta de Leyenda, permite configurar cuales son las capas visibles en el mapa.

4.7.1.2 Ingreso de usuario.

El usuario debe identificarse ingresando un usuario registrado y una contraseña. Eso es fundamental para que el sitio web pueda ofrecer algún tipo de personalización al usuario. Por ejemplo vistas del mapa personalizadas o recordar las planificaciones de rutas realizadas por el usuario. En este prototipo no se incluyen estas posibilidades.

4.7.1.3 Opciones de Navegación.

Las opciones de Navegación permiten acceder a las herramientas que tiene disponibles el sitio. La opción de Inicio, permite el acceso a la página principal. La opción de Registrarse, sirve para registrar nuevos usuarios.

La opción de Información de Tráfico muestra en el mapa el nivel de congestión que hay en la ciudad. El nivel de congestión de las calles es determinado mediante la ejecución de la simulación del tráfico de la ciudad (ver el Capítulo 5). En la Figura 4-13 puede verse que las calles de la ciudad tienen distintos colores dependiendo de la velocidad que se alcanza en cada tramo de la calle. El color rojo indica que la velocidad que puede alcanzar un vehículo en ese tramo es menor a 17 kilómetros por hora, es decir hay un nivel de congestión alto. El color amarillo indica un nivel de color medio, la velocidad alcanzada está entre 18 y 36 kilómetros por hora. El color verde indica un nivel normal de circulación, la velocidad alcanzada es superior a 37 kilómetros por hora. En la Figura 4-13 puede verse que hay calles en las que no hay ninguno de estos colores (las de más abajo). Eso es debido a que la simulación solo toma una sección del mapa de Montevideo para simular. En la imagen las calles que están en blanco indican que no estaban incluidas en la zona simulada. La zona simulada es configurada por el usuario.



Figura 4-13. Información de congestión.

La opción Rutear permite realizar planificaciones de ruta. Primero se ingresan los nombres de las intersecciones de las calles de origen y destino. Luego el sistema

valida que el cruce de calles sea correcto y habilita la opción de realizar la planificación. Además presenta la opción de utilizar información de tráfico para que se retorne la ruta en que se demora menos tiempo en llegar al destino o sino la ruta que recorre menos distancia. En la Figura 4-14 se muestra el paso en que se realiza una planificación desde la esquina de Yaro e Isla de Flores hacia Galicia y Minas.

The screenshot shows the 'Viajero Informado' application interface. On the left, there is a navigation menu with options: Inicio, Registrarse, Info Tráfico, Rutear, and Simulador. The main area contains a form for route planning. The origin is set to 'YARO' and 'ISLA DE FLORES', and the destination is 'GALICIA' and 'MINAS'. There is a checkbox for 'Usar información de tráfico' which is currently unchecked. A 'Rutear' button is at the bottom.

Figura 4-14. Configuración de la planificación de una ruta.

En la Figura 4-15 se muestra a la izquierda la planificación de la ruta que recorre menor distancia (en color rojo) y a la derecha la que demora menos tiempo en llegar (en color azul). En la Figura de la derecha se muestra además el nivel de congestión de las calles. Si se comparan las rutas puede observarse que son diferentes entre sí, debido a que se seleccionaron criterios de optimización distintos al calcular las rutas.

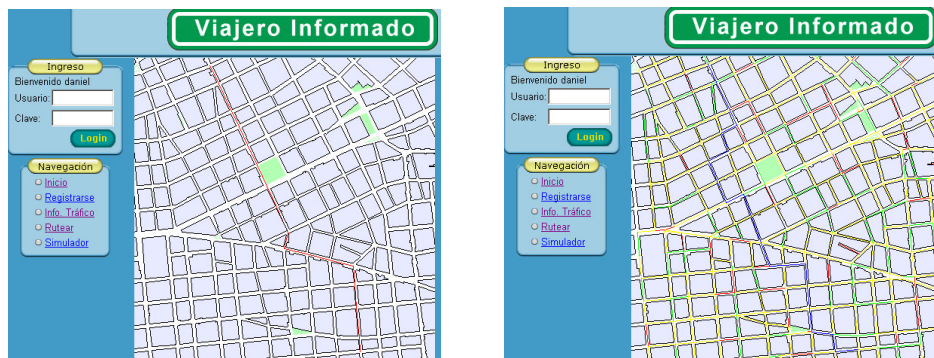


Figura 4-15. Dos planificaciones de un misma ruta usando criterios diferentes.

Finalmente, la opción Simulador permite navegar por el mapa de la ciudad y obtener la extensión de la vista actual del mapa. En el simulador uno de parámetros configurables es justamente la extensión del mapa, es decir la zona de la ciudad que encierra a todas las calles que van a ser utilizadas para la simulación. El simulador es un programa independiente de Viajero Informado, por lo que ese parámetro luego debe configurarse desde el archivo de inicio del simulador, véase el Capítulo 5. Esta opción se presenta en Viajero Informado para reutilizar la capacidad de navegación

del sitio. Sino sería necesario reprogramar esta funcionalidad desde el simulador. En la Figura 4-16 se puede ver una imagen de esta funcionalidad donde se ve el mapa de la ciudad y la extensión que el corresponde, en este caso la extensión es: 464731.5 6137064.850078 466698.718750 6139034.149922.



Figura 4-16. Extensión del mapa.

4.7.2 Implementación.

Para el desarrollo de Viajero Informado se utilizó Genexus versión 7.5. Los programas fueron generados en el lenguaje Java para la Base de Datos PostgreSQL. Para ejecutar las páginas web, se necesita un servidor web que interprete los programas Java y devuelva las páginas. A ese tipo de servidor se le conoce como servidor de servlets y se utilizó uno llamado Resin [64]. El servidor de servlets puede cambiar, sin necesidad de reprogramar la aplicación. Generalmente el servidor seleccionado depende de las dimensiones de la aplicación, la cantidad de usuarios que la acceden y la plataforma del servidor. En este caso Resin es adecuado para el desarrollo de este prototipo ya que no consume demasiados recursos del sistema.

4.7.2.1 Genexus.

Genexus [44] es una herramienta de desarrollo de aplicaciones de Base de Datos críticas, creada por una empresa uruguaya llamada ARTech. Genexus genera programas a partir de un lenguaje de desarrollo propio. El usuario puede seleccionar el lenguaje y la Base de Datos en la cual quiere que sus aplicaciones sean generadas. El manejo de la Base de Datos se realiza en forma automática, generándose el diseño de las tablas normalizadas en tercera forma normal. Genexus cuenta con objetos propios para desarrollar la aplicación. Estos objetos son: Transacciones que definen la estructura de la Base de Datos y formularios para el ingreso de la información, Workpanles que son formularios para aplicaciones de tipo Windows, Webpanels que

definen páginas web, Procedimientos que permiten realizar operaciones de acceso a la Base de Datos u operaciones de procesamiento y Reportes que permiten imprimir listados. Genexus utiliza una metodología de desarrollo de prototipos incrementales con la cual se facilita el proceso de desarrollo de la aplicación final.

4.7.2.2 Módulos desarrollados.

A continuación se describen todos los módulos desarrollados para Viajero Informado.

4.7.2.2.1 Manejo de parámetros del sistema.

Este módulo se encarga de manejar los parámetros del sistema que manejan distintos aspectos de la configuración del sistema y las sesiones del usuario. Esta parametrización permite que se puedan modificar distintos aspectos del comportamiento de los programas, sin necesidad de modificar la programación ya realizada. Por otro lado esta parametrización tiene como objetivo permitir la personalización de las funcionalidades en Viajero Informado. Cuando un usuario ingresa al sitio algunos de los parámetros son copiados en la sesión para que el usuario los modifique. Por ejemplo los parámetros relacionados con las capas geográficas permiten configurar si el usuario quiere ver esa capa en el mapa o no. Otro parámetros permiten configurar capas geográficas de puntos, donde el usuario puede definirlos. De este modo el usuario pueden definir puntos que le interesen en la ciudad, por ejemplo la ubicación de su casa o su trabajo y usarlos para planificar rutas. Esta funcionalidad fue implementada cuando se utilizaban Shapefiles para el manejo de la información geográfica. Cuando se comenzó a utilizar PostgresSql y PostGIS, no alcanzó el tiempo para migrar esta aplicación para que ingresara los puntos en la Base de Datos en lugar de los Shapefiles, por lo tanto fue quitada de lo que es Viajero Informado actualmente. En el Apéndice B se describen en detalle todos los parámetros utilizados.

Procedimientos.

Procedimiento	Descripción
SPActUsrLayer	Activa las capas de usuario
SPCpyCategory	Copia una categoría
SPDeleteSesion	Elimina una sesión
SPGetLyrCat	Retorna la categoría de una capa geográfica
SPGetNmeVal	Recupera el nombre y el valor de un parámetro
SPGetParVal	Recupera el valor de un parámetro
SPGetWebUser	Recupera el usuario de la sesión
SPNewSesion	Crea una nueva sesión
SPSetParVal	Inicializa el valor de un parámetro
SPSetWebUser	Inicializa el usuario
SPUsrLyrActive	Indica si la capa está activa
SPValidateSesion	Valida la sesión

Web Panels.

Estas páginas web implementan las herramientas administrativas para la configuración de los parámetros del sistema.

Web Panel	Descripción
SPBO001	Aplicación principal
SPBO002	Trabajar con parámetros
SPBO003	Trabajar con categorías de parámetros
SPBo005	Copiar categorías
SPReporteVal	Lista los parámetros

4.7.2.2.2 Manejo de Javascripts.

Este módulo se encarga del manejo de los Javascripts utilizados para la interacción con la imagen del mapa con jBox.

Procedimientos.

Estos procedimientos implementan operaciones para recuperar los JavaScripts y instanciarlos como corresponda.

Nombre	Descripción
KNScGetScript	Recupera un Script
KNScRepWord	Reemplaza una palabra en el Script

Workpanels

Implementa la aplicación de administración de los JavaScripts.

Nombre	Descripción
KNScBo01	Trabajar con Scripts

4.7.2.2.3 Comunicación con el Servidor de Mapas.

Este módulo implementa la interacción con el servidor de mapas. El módulo consiste en procedimientos que cumplen distintas funciones para lograr la comunicación con el servidor. Algunos se encargan de crear la URL del servidor de mapas, pasando todos los parámetros en la invocación y otros se encargan de procesar la respuesta.

Procedimientos.

Nombre	Descripción
CGIParam2	Concatena parámetros en una URL
MSGetQueryUrl	Construye una URL en modo query
MSGetScript	Retorna los Scripts utilizados por el mapa
MSGetUrl	Construye una URL para comunicarse con el Servidor de Mapas
MSGetUrl2	Construye una URL para comunicarse con el Servidor de Mapas, pero especifica el punto donde el usuario hizo clic en la imagen
MSJSAppletImg	Retorna el applet Java utilizado en la imagen del mapa
MSJSMapplet	Retorna el JavaScript utilizado en la imagen del mapa
MSXml5	Realiza la llamada al Servidor de Mapas
MSGetQueryTrafficInfoUrl	Crea la URL para que el Servidor de Mapas muestre la información de tráfico
MSGetUrlForSimulation	Crea la URL para que el Servidor de Mapas muestre la información de la simulación

4.7.2.2.4 Manejo de usuarios web.

Para el manejo de usuarios que acceden a Viajero Informado hay una tabla en la Base de Datos donde se almacena la información relativa al usuario y operaciones para recuperar la información.

Procedimientos.

Nombre	Descripción
WUGetLayer	Retorna las capas de usuario de un usuario
WUGetUName	Retorna el nombre del usuario
WUIsLogIn	Retorna si el usuario tiene una sesión válida
WULogin	Valida si la clave del usuario es correcta
WUNewWUsr	Ingresa un nuevo usuario

4.7.2.2.5 Páginas Web de Viajero Informado.

Las páginas web implementan la interfaz con el usuario de Viajero Informado.

Procedimientos.

Nombre	Descripción
ERBuscarCalle	Busca una calle por el nombre
ERCallRoutingAlg	Invoca al algoritmo de planificación de rutas
ERConfirmEsquina	Confirma si existe la intersección de dos calles
ERGetGeoRouteId	Retorna el identificador de una ruta
ERInsOrigDest	Inserta la esquina origen y destino de una planificación

WebPanels.

Nombre	Descripción
WICabecal	Cabecal de las páginas web
Wilogin	Componente de ingreso al sistema
WIMain	Página principal de Viajero Informado
WIMainSimulador	Página principal para la información de la simulación
WIMap	Maneja el mapa
WIMapaRuteo	Maneja el mapa que muestra las rutas
WIMapaTraficoRuteo	Maneja el mapa que muestra la ruta con la información de tráfico
WIMapSimulador	Mapa que muestra la extensión para la simulación
WIRuteo2	Componente que permite seleccionar las calles para la planificación de la ruta
WIRutMain	Página principal de la planificación de rutas
WISimMain	Página principal de la información de la simulación
WISimMap	Mapa con la información de tráfico
WIVerRuteo	Presenta el mapa con la ruta
WIVerTraficRuteo	Presenta el mapa con la ruta e información de tráfico

4.7.2.2.6 Numerador.

Este módulo sirve para el manejo de los numeradores de los distintos identificadores utilizados.

Procedimientos

Nombre	Descripción
NUMGetNumber	Obtiene un número
NUMSetUltNum	Salva el ultimo número

4.7.2.2.7 Invocación del algoritmo de planificación de rutas.

El programa que ejecuta el algoritmo de planificación de rutas es un programa ejecutable con extensión .EXE. Este programa no tiene interfaz gráfica, es un programa que se ejecuta desde una línea de comando. Recibe seis parámetros:

1. Esquina de origen (dos nombres de calles).
2. Esquina de destino (dos nombres de calles).
3. Identificador de la ruta con la que debe salvar la planificación en la Base Datos.
4. Y o N dependiendo si debe utilizar información de tráfico o no.

Para ejecutar el programa se utiliza el comando de Genexus SHELL que permite ejecutar comandos desde una consola MS-DOS. El comando SHELL es ejecutado en forma sincrónica, de modo que cuando el programa de planificación de rutas termina, las páginas web continúan con las ejecución. Un ejemplo de la invocación al programa de planificación de rutas que realiza Viajero Informado, sería así:

```
C:\progs\graf2.exe "DIECIOCHO DE JULIO" "MINAS" "DIECIOCHO DE JULIO" "EJIDO" 145 Y
```

El nombre del programa de planificación de rutas es graf2.exe y se encuentra localizado en c:\progs. Los parámetros que se le pasan indican que debe buscar una ruta desde la esquina de 18 de Julio y Minas hacia la esquina de 18 de Julio y Ejido, utilizando información de tráfico y salvando la ruta con el identificador 145.

Los objetos Genexus que realizan esta tarea pertenecen al módulo de Páginas Web, en particular el procedimiento *ERCallRoutingAlg* se encarga de hacer la invocación al programa de planificación de rutas.

4.7.2.2.8 Interacción con el mapa.

Para cargar el mapa en la página web, se inserta el código HTML en el webpanel. Este código tiene dos elementos importantes que permiten que aparezca la imagen del

mapa y que sea posible interactuar con ella. Uno es la URL de la imagen con el mapa que creó el Servidor de Mapas. El otro elemento importante es un Javascript que realiza la interacción con jBox y permite que sea posible marcar un punto o un rectángulo haciendo clic en el mapa. Este código Javascript es recuperado de la Base de Datos y se le incorpora la URL con la imagen del mapa. En la Figura 4-17 se ve al mapa de Montevideo y la selección que realiza el usuario es el rectángulo magenta. Cuando el usuario marca un rectángulo o un punto, el Javascript que se encarga de interactuar con jBox carga en variables en la página web las coordenadas de la selección realizada. Estas coordenadas son relativas a las dimensiones de la imagen y no al mapa. Con esas coordenadas y la extensión del mapa que ve el usuario, el Servidor de Mapas puede determinar exactamente cuales son las coordenadas del nuevo mapa. La acción predeterminada cuando el usuario selecciona un rectángulo en el mapa es hacer un zoom a esa zona. En el caso que el usuario marque un punto en el mapa la acción depende de la opción seleccionada por el usuario, puede ser centrar o hacer zoom.

Los objetos Genexus que realizan estas tareas pertenecen a los módulo de Páginas Web, Manejo de Javascripts y comunicación con el Servidor de Mapas.

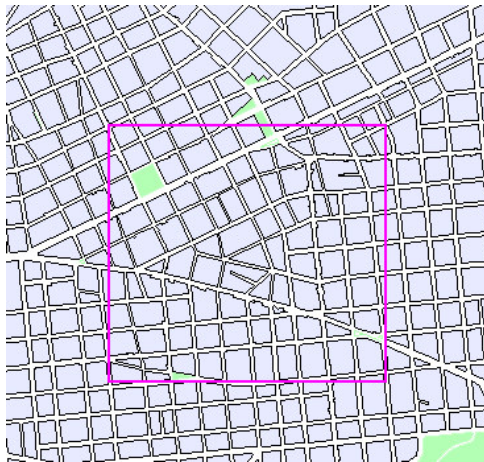


Figura 4-17. Selección en el mapa.

5 Simulación del tráfico de la ciudad.

Los algoritmos de planificación de rutas, al generar una ruta pueden tener en cuenta, las condiciones del tráfico. Dicha información es generada mediante una simulación del tráfico de la ciudad. En este Capítulo se describe el modelo y la implementación de la simulación.

El simulador tiene como objetivo principal proveer información de tráfico de la ciudad, para que el algoritmo de planificación de rutas pueda utilizarla. Como no se tiene disponible información real de tráfico, la simulación es una excelente herramienta para proporcionarla. El desarrollo del simulador permitió investigar en las distintas teorías sobre el tráfico de vehículos e implementar uno de los modelos estudiados.

5.1 Simulación del tráfico de vehículos.

Las simulaciones en computadoras son herramientas muy utilizadas en el área del transporte. Permiten modelar distintos fenómenos observados en la realidad, sin necesidad de hacer experimentos peligrosos para el sistema. Distintos modelos de simulación se comenzaron a desarrollar a partir de la década de 1950 y hasta el día de hoy han evolucionado mucho [65]. Principalmente influenciados por la evolución en las técnicas y la tecnología que se utiliza para obtener la información de tráfico en las calles. Las nuevas tecnologías permiten tomar otras medidas que antes no se tenían disponibles y esto inevitablemente hace que los modelos de simulación deban ser redefinidos para contemplar esta nueva información.

Los modelos de simulación de tráfico se pueden dividir en dos grandes categorías *microscópica* y *macroscópica*, dependiendo del nivel de abstracción que utilicen. En los modelos microscópicos o de micro simulación se modela el tráfico a nivel de cada vehículo. En los modelos macroscópicos se abstrae hasta ver el comportamiento de todo el sistema permitiendo así tener una visión global del comportamiento.

El modelo escogido en este caso es un modelo de micro simulación llamado Modelo de Autos Siguiéndose (MAS) (Car Following Model) [66].

5.2 Modelo de Autos Siguiéndose.

Los MAS o también llamados modelos de seguir al líder, modelan el comportamiento de cada vehículo individual dependiendo del vehículo que lo antecede. La mayoría de los modelos para modelar este comportamiento utilizan ecuaciones diferenciales de segundo orden que relacionan un vehículo con el vehículo que tiene adelante [67]. Sin embargo, para la implementación se seleccionó un modelo que utiliza autómatas celulares, propuesto originalmente por Nagel y Schreckenberg en [66] y descrito también en [68]. Aunque en el artículo original se modela el tráfico en carreteras de alta velocidad con una sola vía de circulación, puede ser adaptado al tráfico de una ciudad del mismo modo que se hace en los trabajos [69], [70], [71], [72], [73] y [74].

5.2.1 Autómatas celulares.

Los autómatas celulares son autómatas que definen: un universo, elementos que habitan en él y reglas que determinan su comportamiento. A medida que el tiempo va pasando, los elementos van evolucionando de acuerdo a las reglas. Uno de los ejemplos más conocidos de autómatas celulares es el Juego de la Vida creado por John Conway [75]. Los autómatas celulares se han aplicado exitosamente en áreas como vida artificial y distintos tipos de modelos de simulación en distintas áreas [75].

En el modelo de autómatata celular utilizado en el Modelo de Autos Siguiéndose, el universo está dado por las calles de la ciudad y los elementos son los vehículos que se mueven siguiendo un conjunto de reglas. El modelo se puede extender fácilmente para representar otros elementos, por ejemplo conductores con distintos perfiles psicológicos, distintos tipos de vehículos, distintas representación de cruces de calles o semáforos (en estos artículos se proponen distintas variaciones a estos elementos: [70], [71], [72] y [73]).

En las siguientes secciones se expondrá primero el modelo para una sola calle y luego se extenderá para todas las calles de una ciudad.

5.3 Modelo para una sola calle.

En esta Sección se describe como fue adaptado el modelo propuesto originalmente por Nagel y Schreckenberg para simular el tráfico en una ruta, al tráfico de una calle de la ciudad. Una calle es representada en el modelo por un arreglo de celdas fijo L_c . Cada celda puede estar ocupada por un solo vehículo o puede estar vacía. Los vehículos solo se mueven en una dirección por las celdas. Cada vehículo tiene una velocidad discreta que va de cero a v_{max} . La velocidad significa la cantidad de celdas que un vehículo puede avanzar en una actualización de tiempo. Los vehículos son insertados en la primera celda de la calle y circula hacia la derecha hasta llegar a la última celda en la que son eliminados. En la Figura 5-1 se presenta un esquema del funcionamiento, el número que está en las celdas corresponde a la velocidad que llevan los vehículos.

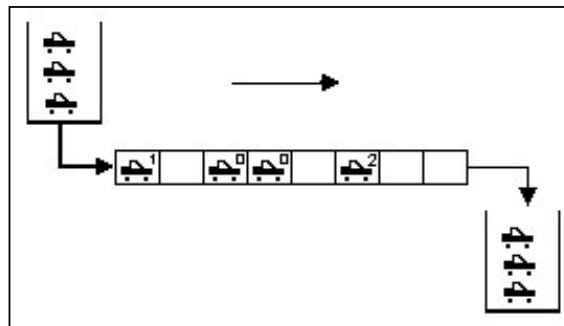


Figura 5-1. Funcionamiento del modelo.

El tiempo de la simulación es discreto e interno a la simulación. La simulación funciona a eventos discretos donde los eventos son los incrementos del tiempo. En cada paso de tiempo se ejecutan reglas que gobiernan el movimiento de todos los vehículos simultáneamente, en una actualización paralela (la actualización secuencial tiene diferentes resultados, según [66]). Estas reglas son llamadas Reglas de Movimiento de Vehículos.

Se definirán algunas funciones para facilitar la comprensión de las reglas de movimiento. Si i es el número identificador de un vehículo entonces, $x(i)$ es la posición dentro del arreglo, $v(i)$ la velocidad actual, $antecesor(i)$ es el identificador del vehículo que i va siguiendo y $espacio_disponible(i) := x(antecesor(i)) - x(i) - 1$ es la cantidad de celdas que quedan vacías entre i y el vehículo siguiente. Las reglas de movimiento de los vehículos son las siguientes:

1. **Aceleración.** Si $v(i) < v_{max}$ entonces, $v(i) := v(i) + 1$.
2. **Desaceleración debido a otros vehículos.** Si $v(i) > espacio_disponible(i)$ entonces $v(i) := espacio_disponible(i)$
3. **Desaceleración aleatoria.** Si $v(i) > 0$ y $número_aleatorio() < P_{des}$ entonces $v(i) := v(i) - 1$.
4. **Movimiento del vehículo.** $x(i) := x(i) + v(i)$.

La regla de aceleración modela la tendencia de los conductores de ir siempre a la máxima velocidad permitida. La desaceleración debido a otros vehículos evita choques con el vehículo que va adelante y que se ocupen celdas ya ocupadas. En el trabajo original de Nagel y Schreckenberg esta regla decreta en uno la velocidad. Sin embargo cuando se extienden estas reglas a una ciudad es necesario permitir mayor flexibilidad en la desaceleración. Esto se debe a que las condiciones de velocidad pueden variar muy rápidamente, por ejemplo, si se dobla una esquina podría encontrarse un vehículo muy próximo entonces sería necesario disminuir la velocidad rápidamente. La desaceleración aleatoria utiliza la función $número_aleatorio()$ (retorna un número generado aleatoriamente) y un parámetro de probabilidad de desaceleración P_{des} para modelar el comportamiento imprevisible del conductor. La regla de movimiento del vehículo efectúa el avance de los vehículos.

La regla de desaceleración es fundamental para lograr una simulación realista del tráfico, sino el comportamiento sería completamente determinístico. La regla modela la fluctuación en la velocidad debido al comportamiento impredecible del conductor o a variaciones en las condiciones externas, por ejemplo el cruce de un peatón.

El modelo de Nagel y Schreckenberg permite reproducir naturalmente el fenómeno de las congestiones espontáneas. En la Figura 5-2 se muestra un diagrama de espacio tiempo obtenido mediante secuencias de fotografías aéreas de una carretera, en la que se puede apreciar la generación espontánea de una congestión [66]. En la Figura 5-2 se pueden apreciar las posiciones de los vehículos en el camino, a medida que pasa el tiempo. En el eje horizontal se grafica la posición que tiene un vehículo en el camino, que se mueve hacia la derecha y en el eje vertical la evolución en el tiempo. El resultado son líneas, que muestran a cada paso del tiempo la posición en la ruta que ocupa un vehículo. La parte más oscura de la figura muestra como se forma una congestión y como se propaga hacia atrás en el camino. Este tipo de comportamiento es modelado por la regla de desaceleración aleatoria.

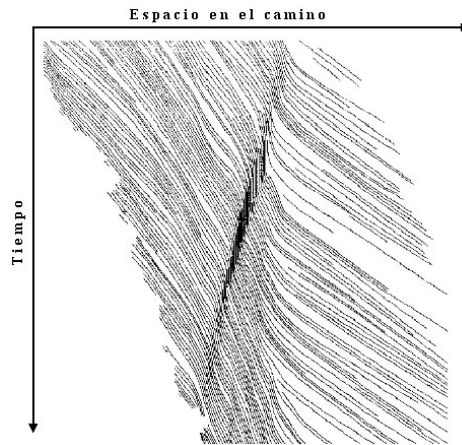


Figura 5-2. Diagrama de espacio tiempo.

A partir de los experimentos realizados, Nagel y Schreckenberg llegan al llamado diagrama fundamental, que muestra la relación entre el flujo y la densidad (en las siguientes secciones se definen estos conceptos), que es verificado con datos obtenidos de la realidad. En la siguiente figura se puede ver la relación entre el flujo y la densidad. Esta relación está indicando que a medida que hay más vehículos en la ruta (aumenta la densidad), el flujo (pasaje de vehículos) va aumentando hasta que llega a un punto máximo. En ese momento se produce una congestión. Debido a la cantidad de vehículos que hay, se comienzan a entorpecer unos a otros disminuyendo el flujo.

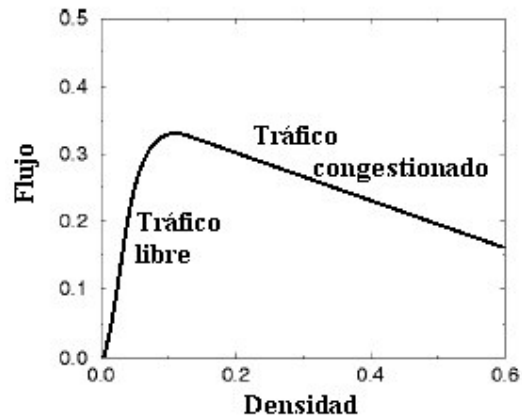


Figura 5-3. Gráfica de Flujo vs. Densidad.

En definitiva Nagel y Schreckenberg presentaron un modelo sencillo y que permite representar el comportamiento del tráfico en carreteras de alta velocidad. Este modelo es suficientemente flexible como para poder adaptarse al tráfico de una ciudad. En la siguiente Sección se explica cómo fue adaptado.

5.4 Extensión del modelo para una ciudad.

El modelo propuesto por Nagel y Schreckenberg fue creado para modelar una sección de una carretera. Para extenderlo a una ciudad es necesario hacerle algunas modificaciones y extensiones. Las modificaciones surgen principalmente de introducir las intersecciones de las calles. Para adaptar el modelo se siguen las ideas propuestas en distintos trabajos que también desarrollan un modelo para una ciudad como: [71], [73] o [74].

5.4.1 Calles

La información de las calles de una ciudad es obtenida a partir de la información de un mapa digital, tal como se explicó en el Capítulo anterior. La cantidad de celdas L_c , que es fija en el modelo original, se extiende en la ciudades de forma proporcional al largo de la calle. Por lo tanto cada calle puede tener una cantidad de celdas distintas. Es posible modelar otros aspectos de las calles, por ejemplo la cantidad de carriles, restricciones de velocidad propias de la calle o semáforos, aunque para la implementación realizada no estaba disponible esa información.

5.4.2 Intersecciones.

Otro punto que debe tenerse en cuenta cuando se adapta el modelo es, como se implementan las intersecciones de las calles. Las intersecciones tienen un comportamiento especial y es necesario agregar reglas de movimiento específicas para manejarlas. En la bibliografía consultada se encontraron dos modos distintos de implementar las intersecciones. La intersección *sin lugar físico*, implica que no hay celdas destinadas al cruce, por lo tanto los vehículos pasan directamente de la calle en la que están, a la siguiente. La otra implementación es mediante *rotondas*, en las cuales los vehículos transitan hasta encontrar la entrada a la calle que quieren acceder. Para la implementación se optó por la intersección sin lugar físico. A continuación se describen los dos modos.

5.4.2.1 Intersección sin lugar físico.

En este caso las intersecciones no tienen un lugar físico, es decir que un vehículo nunca está en una intersección [71]. Lo que se modela es como si las calles fueran continuas y no compartieran el espacio físico con otras calles. En las calles se diferencian las celdas, las primeras celdas se llaman de *entrada* y las últimas de *salida*. Cuando un vehículo llega a alguna de las celdas de salida, puede ser traspasado hacia alguna de las celdas de entrada en alguna de las calles que forman la intersección. En la Figura 5-4 se muestra un esquema del funcionamiento, el bloque representa una pared imaginaria en la calle.

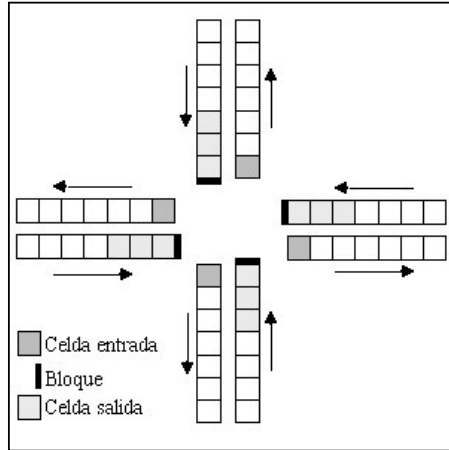


Figura 5-4. Intersección sin lugar físico.

La cantidad de celdas de entrada depende de como quiere modelarse la intersección. Mientras más celdas de entrada tenga la calle, permitirá el pasaje de más vehículos desde la otra calle. Esto es útil para modelar ampliaciones en los carriles de las calles. En la implementación realizada se utilizó solo una celda de entrada. La cantidad de celdas de salida que tiene que haber es igual a v_{max} . De esta manera se asegura que el vehículo no sobrepase nunca el limite de la calle en la que está, en el caso de que no pueda pasar a la siguiente. La selección del vehículo que va a traspasar a la celda de entrada, mantiene el orden de los vehículos, es decir el primero que llega a la zona es el primero que pasa.

El manejo de las intersecciones de este modo requiere reglas de movimiento especiales llamadas de Traspaso de Vehículos. En una actualización del tiempo, primero se ejecutan las reglas de traspaso de vehículos y luego las de movimiento. Las reglas de traspaso que se siguen son:

1. **Traspaso de vehículos.** Si el vehículo más próximo al fin de la calle se encuentra sobre las celdas de salida y $x(i) + v(i) > L_c$ entonces:
 - a. Si alguna de las calles siguientes tiene su celda de entrada vacía, mover el vehículo.
 - b. Sino, si ninguna celda de entrada está vacía entonces, $x(i) := L_c$.

Lo que indica la regla de traspaso es que los vehículos que estén en el área de salida y tengan suficiente velocidad como para cruzar la calle, entonces se los pondrá en la celda de entrada de la siguiente calle, sino quedarán en la última celda de la calle esperando que se libere la celda de entrada. La selección de la calle a donde se mueve el vehículo depende de si el vehículo sigue una ruta fija o es seleccionada al azar.

Es necesario además, hacer una pequeña modificación a la cuarta regla de movimiento. El problema es que si un vehículo se encuentra en las celdas de salida y aumenta su velocidad, podría ser que su nueva posición supere el largo máximo de la celda. Por lo tanto la cuarta regla se modifica de la siguiente manera:

4. **Movimiento del vehículo.** Si $x(i) + v(i) < L_c$ entonces $x(i) := x(i) + v(i)$ sino $x(i) := L_c$.

5.4.2.2 Intersección con rotondas.

En este caso las calles que se interceptan son conectadas por una rotonda. La rotonda esta formado por un arreglo de celdas que conecta todas las calles. En la Figura 5-5 se ve una adaptación de un figura presentada en el trabajo de Dupuis y Chopard [73]. En el ejemplo, las flechas indican el sentido de las calles. Los vehículos circulan en las rotondas en sentido antihorario. En el trabajo de Chopard y otros [72] las rotondas tienen una celda por cada calle que se cruza, mientras que en el trabajo de Dupuis y Chopard [73] la cantidad de celdas depende de la cantidad de calles que se conectan en la rotonda.

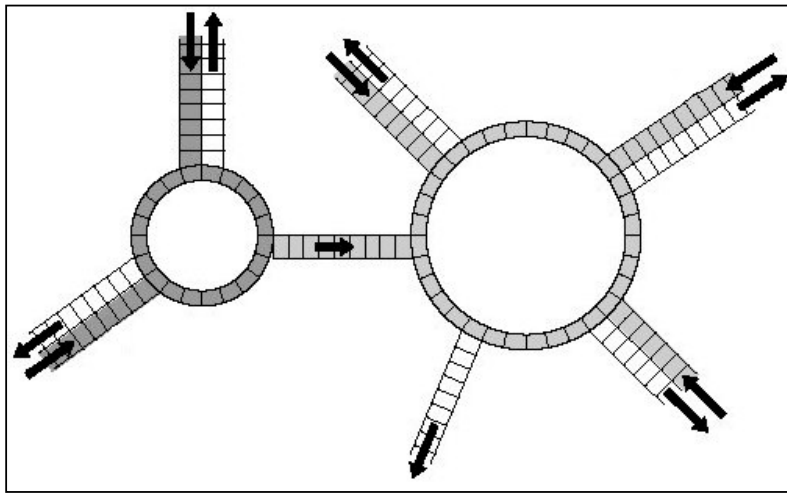


Figura 5-5. Calles con rotondas.

Este enfoque tiene como ventaja frente al anterior que las reglas de traspaso de vehículos no son necesarias. Las mismas reglas de movimiento son utilizadas en las rotondas ya que todas las calles están conectadas por celdas. Además, se le da prioridad a los vehículos que están en la rotonda, esto da un modo más real de resolver problemas de concurrencia en situaciones donde todos los vehículos se mueven sincronizados. La desventaja de este enfoque es que aumenta la cantidad de celdas, que es el aspecto más crítico en términos de desempeño de la simulación.

5.4.3 Escala del modelo.

Para que el modelo refleje la realidad de la ciudad de Montevideo, los parámetros de la simulación son ajustados de acuerdo a las características de la ciudad. En el trabajo original de Nagel y Schrenckenberg [66] se presentan tres posibles formas de escalar el modelo. Un modo es utilizando el diagrama fundamental presentado anteriormente en la Figura 5-2, otro modo es ajustando la capacidad máxima del pasaje de vehículos en la ruta y el tercer modo, que es el seleccionado, utiliza la relación entre velocidad, distancia y tiempo. Los dos modos descartados son específicos para el tráfico de una

ruta y sería muy difícil utilizarlos para un modelo a mayor escala como el de la ciudad, donde las calles tienen distintas características, ver [66] y [76]. A continuación se describe la forma de escalar el modelo.

En la simulación hay tres parámetros para escalar el modelo:

- Cantidad de metros de largo que tiene una celda.
- Velocidad máxima a la que van los vehículos.
- Relación entre el tiempo de la simulación y el tiempo real.

Para el ajuste se considera que la velocidad máxima a la que se puede conducir en la ciudad es aproximadamente de 75 kilómetros por hora, que es la velocidad permitida en algunas calles, principalmente sobre las ramblas costeras. Otro parámetro que se fijó es mantener una unidad de tiempo de la simulación aproximadamente igual a un segundo de la realidad. Con estas dos condiciones se determinaron los siguientes valores de los parámetros:

- Una actualización de tiempo en la simulación corresponde a un segundo real.
- Largo en metros de una celda, 5 metros.
- Velocidad máxima, 4 celdas por segundo.

La ecuación para determinar estos parámetros surge de plantear que el tiempo es igual a la distancia dividido la velocidad. La ecuación da los siguientes resultados:

$$\frac{5 \frac{m}{celda} * 4 \frac{celda}{actualización}}{\frac{72 \frac{s}{3.6 m}}{s}} \approx 1 \frac{actualización}{s}$$

Si el largo de la celda es 5 metros, los vehículos se desplazan a 4 celdas por cada actualización y si se considera una velocidad 72 kilómetros por hora (72/3.6 metros por segundo), se obtiene que una actualización de tiempo de la simulación es equivalente a un segundo real. La relación que hay entre la velocidad en la simulación y la velocidad de los vehículos en la realidad se muestra en la siguiente tabla:

Celdas por segundo	Kilómetros por hora
4	72
3	54
2	36
1	18
0	0

Entonces según la tabla, cuando en la simulación un vehículo avanza a cuatro celdas por segundo, en la realidad el vehículo esta conduciendo a 72 kilómetros por hora.

5.5 Medidas.

En la teoría de tráfico existen diferentes medidas que sirven para estimar la utilización de las calles y carreteras. Algunas de ellas son:

- Flujo, es la cantidad de vehículos por unidad de tiempo.
- Velocidad, distancia por unidad de tiempo.
- Tiempo de viaje sobre un segmento del camino conocido.
- Ocupación, es el porcentaje de tiempo que un segmento del camino es ocupado por vehículos.
- Densidad, es la cantidad vehículos por unidad de distancia.
- Tiempo entre el pasaje de un vehículo y otro.
- Distancia entre un vehículo y otro.

Los procedimientos utilizados para obtener estas medidas han evolucionado en los últimos 60 años debido: al creciente interés en el flujo del tráfico, los avances tecnológicos y la proliferación de carreteras de alta velocidad. Algunos métodos de observación son [65]:

1. Medir en un punto. Con este método se obtiene información de un punto específico de la ruta. Para tomar la medida se utilizan métodos como radares o cámaras de televisión.
2. Medir sobre una sección corta del camino, por ejemplo 10 metros. En este método se utiliza generalmente dos medidores del tipo que miden en un punto separados.
3. Medir sobre una sección más larga, por ejemplo 0.5 kilómetros. En este caso las medidas son tomadas de fotografías aéreas o de cámaras montadas a gran altura. Este método de medición se utiliza principalmente en carreteras.
4. Un vehículo moviéndose en el tráfico. En este caso la información es registrada por un vehículo que va conduciendo en el tráfico.
5. Muestras a gran escala, obtenidas simultáneamente de varios vehículos como parte de la infraestructura del Sistemas de Transporte Inteligentes. En este caso los vehículos están equipados con dispositivos que permiten comunicar a una central su posición y velocidad. En algunos casos se instalan en las calles dispositivos de reporte que permiten obtener la información del vehículo cuando pasa. En otros sistemas más avanzados se puede obtener la información del vehículo sin importar donde este.

Las variables que podrán estimarse, dependen del método que se utilice. En el reporte de Gartner y otros [65] se detalla cuales son las variables que pueden estimarse, dependiendo del método de medición utilizado.

Para la simulación podría utilizarse cualquiera de estos métodos para obtener medidas. Se seleccionó el método de medir sobre un punto, ya que ofrece información detallada de las calles. Si se utilizara mediciones mediante vehículos como proponen los métodos 4 y 5 podría ocurrir que sobre una calle no se tenga información y sea necesaria estimarla.

Si bien se censa en un punto, en realidad se censa en un intervalo de celdas de tamaño v_{\max} . Esto se hace para asegurarse que todos los vehículos que pasen por el censor sean detectados. Los sensores son colocados en la mitad de las calles tal como se recomienda en [6]. Colocando el censor en un lugar cercano a la mitad de la calle se evitan las distorsiones por la disminución de la velocidad que realizan los conductores al atravesar un cruce.

A continuación se definen las variables sobre las que se toma información.

5.5.1 Densidad.

La densidad del sistema ρ se define como la cantidad de vehículos por unidad de distancia, en un lapso de tiempo. En este caso la densidad será la cantidad de vehículos por celda. La densidad u ocupación $\bar{\rho}^T$ en una celda fija i promediada en un período de tiempo T se define como:

$$\bar{\rho}^T = \frac{1}{T} \sum_{t=t_0+1}^{t_0+T} n_i(t)$$

$$\text{Donde: } n_i(t) = \begin{cases} 0 & \text{si el sitio } i \text{ está vacío} \\ 1 & \text{si el sitio } i \text{ está ocupado} \end{cases} \quad \text{en tiempo } t$$

Esta definición sirve para un sola celda, para adaptarla a un conjunto de celdas, se cuenta la cantidad de vehículos que hay en ellas. Es decir se utiliza la siguiente definición:

$$n_Z(t) = \begin{cases} 1 & \text{si hay alguna celda ocupada en } Z \\ 0 & \text{en otro caso} \end{cases} \quad \text{en tiempo } t$$

5.5.2 Flujo.

Otra medida utilizada es el flujo q , que es la cantidad de vehículos, por unidad de tiempo. El flujo promedio entre la celda i e $i+1$ se define como:

$$\bar{q}^T = \frac{1}{T} \sum_{t=t_0+1}^{t_0+T} n_{i,i+1}(t)$$

Donde $n_{i,i+1}(t) = 1$ si se detecta el pasaje de un vehículo entre la celda i e $i+1$ en el instante de tiempo t .

Para utilizar la definición en el intervalo que utiliza el sensor, lo que se hace es conservar las posiciones de los vehículos desde la última actualización y compararlo con la situación en la siguiente actualización del tiempo. Como los vehículos tienen identificadores únicos, entonces es posible detectar cada movimiento comparando las dos situaciones. La definición queda entonces:

$$n_z(t) = \begin{cases} 1 & \text{si hay algún movimiento de vehículos en } Z \\ & \text{en tiempo } t \\ 0 & \text{en otro caso} \end{cases}$$

5.5.3 Velocidad.

La velocidad a la que circulan los vehículos es un dato importante para medir el nivel de congestión que existe. Para eso se define la *velocidad media en atravesar la calle* o velocidad prevaleciente, que es la velocidad a la que se espera conducir en esa calle. Este valor varía dependiendo del tiempo. Si no pasan vehículos, entonces la velocidad de circulación va a aumentar hasta la velocidad máxima permitida. A medida que circulen vehículos por la calle, la velocidad que se puede esperar alcanzar para atravesar la calle, será similar a la velocidad que tenga el resto de los vehículos. La velocidad de cada vehículo que pasa por la calle es obtenida por el sensor instalado en esa calle. Esta medida es una estimación, ya que supone que la velocidad que lleva un vehículo al pasar por el sensor se mantiene a lo largo de toda la calle.

Para implementar esta medida se utilizó un arreglo que mantiene los registros de velocidad de los vehículos que han pasado últimamente. La cantidad de vehículos que se recuerda es limitada y es proporcional a la cantidad de celdas que tiene la calle. El registro se comporta en forma circular, es decir que cuando todos los registros de velocidad se ocuparon, los más antiguos en ingresar son reemplazados. Cuando se quiere conocer el valor de la medida, se hace el promedio de las velocidades que hay en el registro. Si no pasa ningún vehículo durante un periodo de tiempo configurable, por la zona de medición, se registra como si hubiese pasado un vehículo con velocidad máxima. De este modo se refleja la situación en que disminuye la cantidad de vehículos transitando y la velocidad a la que se puede circular aumenta. En el trabajo de Esser y Schreckenber [74] se implementa este método pero aplicado al tiempo que se demora en atravesar la calle, tal como se describe en la siguiente Sección.

5.5.4 Tiempo.

El tiempo que se demora en atravesar una calle es otra medida que se toma. Esta medida es obtenida dividiendo el largo de la calle entre la velocidad media en atravesar la calle. Esta medida es utilizada luego por el algoritmo de planificación de rutas para planificar recorridos tomando en cuenta el camino por el cual se circula más rápido.

5.5.5 Censores.

Los sensores son los componentes de la simulación encargados de tomar las medidas de flujo, densidad, velocidad predominante y tiempo estimado en atravesar la calle. Para eso se coloca uno en cada calle que se desea tomar mediciones. El sensor examina v_{\max} celdas de la calle y recoge la información. Si encuentra vehículos en las celdas, registra sus velocidades.

En la implementación del simulador se pusieron sensores en todos los tramos de las calles. Esto genera una situación artificial, en el sentido de que es muy poco probable que en un sistema real se pueda tener la información disponible de todos los tramos de todas las calles. Por otro lado el algoritmo de planificación de la ruta, necesita obtener la información completa de todas las calles para poder funcionar correctamente. En un sistema real, es probable que se complete la información que no está disponible con valores por defecto o estimados a partir de los conocidos. En este caso, como no se tienen tales valores, entonces se optó por poner sensores en todas las calles.

5.6 Vehículos.

Los vehículos tienen todas las mismas características, excepto por el valor del identificador que llevan. Es decir, todos circulan con las mismas restricciones de velocidad, ocupan una sola celda y siguen las mismas reglas para desplazarse. El modelo es suficientemente flexible como para permitir la incorporación de distintos tipos de vehículos. Por ejemplo autobuses o camiones, que ocupan mayor cantidad de celdas y circulan a distintas velocidades. También se pueden modelar los conductores con distintos perfiles psicológicos, por ejemplo conductores con tendencia a ir a la velocidad máxima permitida u otros más prudentes.

5.6.1 Controladores.

La cantidad de vehículos en la ciudad varía dependiendo de distintos factores. Por ejemplo, el día de la semana, la hora, las inclemencias del tiempo o acontecimientos especiales en la ciudad. Para reflejar esa variación se incorpora en la simulación otro componente llamado *controlador*. Hay dos tipos de controladores, los que se encargan de generar nuevos vehículos y los que los eliminan. Si se tuviera algún tipo de información sobre la cantidad de vehículos que circulan, los controladores pueden actuar haciendo que la simulación funcione de acuerdo a esas condiciones.

Los controladores actúan de distinta manera dependiendo de si están encargados de insertar vehículos o eliminarlos. Si el controlador debe insertar un vehículo, entonces verifica que la celda donde debe insertarlo no esté ocupada. Si la celda está libre, entonces inserta un vehículo nuevo y le pone una velocidad seleccionada al azar. Para eliminar un vehículo el controlador debe verificar si hay un vehículo en la celda en la que actúa, si comprueba que existe un vehículo, entonces lo elimina.

Los controladores pueden actuar en la simulación de tres formas posibles dependiendo de la configuración que seleccionó el usuario:

- **Probabilidades.** Este modo utiliza dos parámetros, uno indica la probabilidad de que se inserte un vehículo y el otro la probabilidad de que se elimine un vehículo. En cada actualización del tiempo, cada controlador utiliza estos parámetros para determinar si debe actuar o no.
- **Densidad del sistema.** La densidad del sistema es un valor macroscópico de densidad que debería tener todas las calles del sistema. En la simulación se configura un valor de densidad y una tolerancia máxima y los controladores actúan manteniendo la densidad de cada calle en ese rango de valores. Se generan vehículos si la densidad de la calle es menor que la densidad del sistema sino los eliminan.
- **Flujo del sistema.** Este método es similar al anterior pero utiliza el flujo en lugar de la densidad. Si el flujo de una calle es menor que el flujo del sistema, entonces se generan vehículos, sino se eliminan.

Se pusieron dos controladores por cada tramo de las calles de la ciudad, uno genera vehículos y el otro los elimina. Inicialmente estaban colocados en la misma posición para todas las calles. Pero se observó que cuando el controlador está en un período en el que genera muchos vehículos (por ejemplo cuando actúa por probabilidades con un valor muy alto de inserción), los vehículos que circulan por esa calle quedan trancados en la celda anterior en la que el controlador genera los nuevos vehículos. Como los controladores estaban posicionados en la misma celda para todas las calles, entonces todas las calles de la ciudad presentaban la misma situación generando una situación poco real. Si bien la situación no puede remediarse porque ese es el comportamiento esperado, para atenuar el impacto, los controladores son generados en lugares seleccionados al azar.

5.6.2 Rutas de los vehículos.

Para modelar el comportamiento del tráfico de una ciudad debe tenerse en cuenta algún tipo de información sobre las rutas que siguen los vehículos. Generalmente esta información viene dada por Matrices de Origen y Destino [17]. Dichas matrices almacenan información sobre la cantidad de vehículos por hora que van de una zona a otra de la ciudad. Lo que se hace es que los vehículos se dirijan de los orígenes a los destinos mientras que los controladores van aumentando o disminuyendo el número de vehículos según los valores de la matriz [70]. Otra opción es que cierto número de vehículos sigan una planificación de la ruta, simulando que tienen información en tiempo real y otros que realicen un camino aleatorio [74].

Para la implementación no se contó con información de Matrices de Orígenes y Destinos, ni información sobre la cantidad de vehículos que circulan. Por lo tanto se deja que los vehículos circulen libremente por la ciudad, sin ningún tipo de comportamiento predeterminado. Al llegar a una esquina, los vehículos seleccionan una nueva calle que tenga lugar disponible para que puedan cruzar. Si ninguna calle del cruce tiene lugar, entonces el vehículo queda detenido hasta que pueda cruzar. Si el vehículo seleccionara una calle para cruzar y quedara detenido hasta que se libera la calle que seleccionó, podría causar un bloqueo. Estas situaciones en la vida real no ocurren, porque las personas no utilizan reglas estrictas de circulación. En la Figura 5-6 se muestra una situación donde se da un bloqueo, las flechas indican el destino seleccionado por el vehículo.

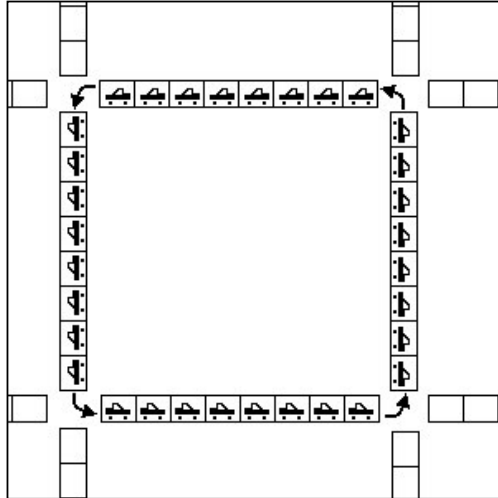


Figura 5-6. Bloqueos.

5.7 Determinación del estado estable

El estado estable de una simulación se refiere a un periodo de tiempo en el cual el comportamiento de los valores de las variables se mantienen razonablemente parecido durante el período de tiempo. En [77] se define el estado estable de una simulación cuando las colas mantienen el mismo comportamiento durante el periodo de tiempo en consideración. Para este modelo de simulación sería bueno poder determinar cuando se alcanza un periodo estable. En la literatura consultada no se sugería ningún criterio que pudiera aplicarse a toda la ciudad. Los criterios que se mencionan son aplicados para el tráfico en carreteras, que tiene distintas características que el de una ciudad. Por lo tanto no hay evidencia clara de que el tráfico de una ciudad sea un sistema estable, sin embargo en este trabajo se le exige a la simulación que cumpla ciertas condiciones de estabilidad. A continuación se describen algunos criterios descritos en la literatura consultada y se describe uno creado especialmente para poder aplicarlo al caso de la ciudad.

En el trabajo de Nagel y Schreckenberg [66], no se menciona una regla particular para determinar cuando la simulación alcanzó el estado estable, simplemente, dejan pasar suficiente tiempo. Luego de ese tiempo se supone que la simulación alcanza el estado estable y toman las mediciones. Dejan pasar $10 \cdot L$, actualizaciones de tiempo, donde L es el largo de la calle.

Por otro lado en Gartner y otros [65] se mencionan dos tipos de estabilidad en las ecuaciones de movimiento. Uno de los criterios está dado desde el punto de vista del comportamiento de un vehículo que sigue a otro. Es decir cuando dos vehículos, que viajan por una calle, uno adelante y otro atrás tienen un comportamiento estable. Este criterio fue creado principalmente para aplicarlo en simulaciones de carreteras donde hay un solo carril, muy largo y donde los vehículos no pueden pasarse unos a otros. Es normal en una ciudad que los vehículos tomen caminos distintos y que las condiciones en la nueva calle no sean las mismas que en la calle anterior. Por otro

lado, el costo computacional de evaluarlo en una ciudad con miles de vehículos sería muy grande. El otro criterio describe la estabilidad de un conjunto de vehículos que circulan por la misma calle por la propagación de las fluctuaciones que experimentan los vehículos en la velocidad. Este criterio está pensado principalmente para simulaciones en carreteras, en las que hay una sola calle y que tiene una gran cantidad de celdas. Sería difícil aplicarlo a una ciudad por que las calles son mucho más cortas, hay pocas celdas para determinar la propagación de la fluctuación y el comportamiento de los vehículos es más impredecible ya que pueden doblar y tomar otras calles.

Como ninguno de estos criterios considera la situación de estabilidad en toda la ciudad, fue necesario crear un criterio particular para contemplar esa situación. Se considera que la simulación está en estado estable si la variación de vehículos se mantiene constante con respecto a la cantidad total de vehículos en promedio durante el tiempo. La variación de vehículos se refiere tanto a inserciones como a la eliminación. La cantidad de vehículos en la ciudad está asociada a los controladores que insertan o eliminan vehículos dependiendo del modo en que fue configurada la ejecución, para alcanzar un valor de flujo o densidad determinado. Entonces, alcanzar el estado estable indica que el flujo o la densidad se mantienen también estables. En la implementación la cantidad de tiempo que tiene que mantenerse estable y los valores de la variación de vehículos son configurables por el usuario.

5.7.1 Tratamiento de callejones.

Los callejones presentan un caso interesante al contabilizar la cantidad de variaciones en la cantidad de vehículos. Como las calles en la simulación tienen dos sentidos, en un callejón ocurren dos situaciones distintas. La calle que lleva hacia el callejón tiende a llenarse de vehículos que no pueden circular ya que la calle no tiene salida y por lo tanto quedan detenidos. Eventualmente, cuando actúa el controlador y elimina un vehículo, todos los que están detrás de él avanzan una posición. En este tipo de calles el valor de densidad es muy alto, ya que todo el tiempo la calle esta ocupada pero el flujo es muy bajo porque los vehículos no circulan.

En las calles que ingresan desde el callejón a la ciudad, prácticamente no hay vehículos. Como la calle no tiene ninguna conexión que le permite el ingreso de nuevos vehículos, los únicos vehículos que circulan por esas calles son los que crea el controlador. En estas calles el flujo y la densidad tienen valores muy bajos. En la Figura 5-7 se presenta esta situación.

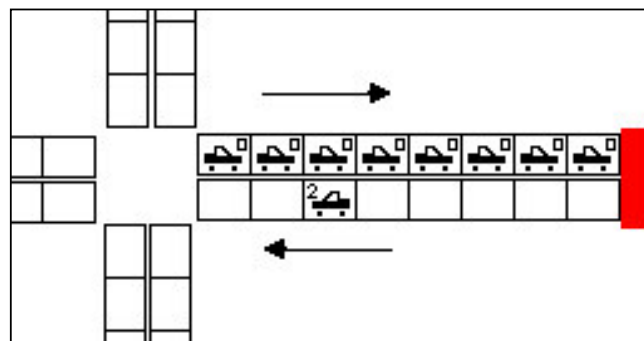


Figura 5-7. Situación normal en un callejón.

Con estos valores de flujo y densidad tan extremos, normalmente los controladores tienden a eliminar vehículos de la calle que va hacia el callejón y a insertar en la calle que sale. Por el modo en que se selecciona una zona de la ciudad para simular, es muy común que una gran cantidad de calles normales, se transformen en callejones en la simulación. Para evitar que el gran número de callejones afecte la determinación de estado estable, los callejones no son tenidos en cuenta al calcular la cantidad de variaciones en la cantidad de vehículos. En la implementación el usuario configura cuales son los callejones.

5.8 Implementación.

Para implementar el simulador se utilizó el lenguaje de programación C++. Se reutilizaron algunos componentes desarrollados para los algoritmo de planificación de rutas como las clases para acceder a la Base de Datos o la implementación del grafo, aunque luego fue modificada. El simulador tiene tres componentes importantes que pueden verse en la Figura 5-8. El componente central es el Motor de la Simulación que es el encargado de ejecutar todo lo que tiene que ver con la simulación. Sobre ese componente está desarrollado una capa de Visualización que permite ver el comportamiento de los vehículos en la ciudad. Por otro lado, la Base de Datos es utilizada para obtener la información de la red de calles de la ciudad y para almacenar la información generada por la simulación. En las siguientes secciones se describen los detalles de cada uno de estos componentes.

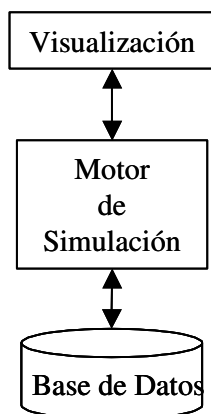


Figura 5-8. Arquitectura del Simulador.

5.8.1 Base de Datos.

En la Base de Datos, como se vio en el Capítulo anterior, están almacenadas las capas de información que son de utilidad para el simulador. El simulador utiliza dos capas, la capa de ejes, de donde saca información de las calles y la capa de manzanas, que utiliza en la visualización.

El simulador utiliza las calles para generar un grafo tal como se explicó en el Capítulo anterior. El usuario selecciona que sección del mapa de Montevideo desea cargar

marcando un cuadrado sobre el mapa en Viajero Informado. De ahí se saca la extensión del mapa que quiere cargar y el simulador recupera todas las calles de esa zona y arma un grafo. La diferencia con el grafo utilizado por los algoritmos de planificación es que este grafo, carga todas las calles de la zona. Otra diferencia que tiene este grafo es que las aristas son dirigidas. Esto se debe a que para que los vehículos circulen las calles deben tener un sentido. Como no se cuenta con la información de flechamientos, entonces todas las calles debieron ser creadas con ambos sentidos de circulación. De este modo se asegura que todos los puntos de la ciudad estén conectados y no se presenten secciones a las cuales no se puede acceder o de las que no se puede salir. En cada una de las aristas se almacenan los arreglos de celdas que contienen a los vehículos.

Luego que el simulador ejecutó y tiene datos de tráfico, se genera una capa geográfica con esa información, que es almacenada en la Base de Datos. La capa tiene datos de velocidad prevaleciente, tiempo que se demora en atravesar la calle, flujo y densidad sobre cada una de los tramos de las calles. Esta información es utilizada por uno de los modos de operación del algoritmo de planificación de rutas.

Antes de crear la capa geográfica con la información del tiempo que se demora en atravesar una calle es necesario adecuar el tiempo al largo real de la calle. La cantidad de celdas que tiene una calle es proporcional a su largo, pero si se suma el largo de las celdas, nunca es exactamente igual. Por otro lado si la cantidad de celdas que le corresponden a una calle es menor que la velocidad máxima, la cantidad de celdas es extendida a la velocidad máxima más dos. De este modo se asegura que los vehículos estarán al menos una unidad de tiempo en cada calle. Para ajustar el tiempo que se demora en atravesar la calle al largo real de la calle y no al largo determinado por la cantidad de celdas se realiza el siguiente calculo para ingresarlo en la Base de Datos:

$$\textit{Tiempo Real} = \textit{Largo Real} * \textit{Tiempo Estimado} / \textit{Largo Celdas}$$

Tiempo Real es el tiempo que demora el vehículo en atravesar la calle, *Largo Real* es el largo verdadero de la calle, *Tiempo Estimado* es el tiempo estimado por la simulación en atravesar la calle y *Largo Celdas* es el largo que tiene la calle si se suman todas las celdas que tiene realmente.

5.8.2 Motor del simulador.

El motor del simulador es el componente que se encarga de ejecutar y coordinar los aspectos de la simulación relacionados con el movimiento de vehículos y las acciones de los controladores y sensores. Para eso el motor ejecuta un algoritmo iterativo en donde cada iteración se corresponde a una actualización del tiempo de la simulación. Las acciones que ejecuta el motor son las siguientes:

1. tiempo := tiempo + 1.
2. Activar los controladores.
3. Reglas de traspaso de vehículos.
4. Reglas de movimiento.
5. Activar los sensores.

Inicialmente el algoritmo incrementa el tiempo generando una nueva actualización. Luego activa los controladores, insertando o eliminando vehículos según corresponda. En los siguientes pasos se ejecutan primero las reglas de traspaso de vehículos y luego las de movimiento, generando así el movimiento de los vehículos. Finalmente se activan los sensores y se toman las estadísticas.

De este modo la simulación logra mover todos los vehículos de la ciudad, tomar las estadísticas y generar o eliminar vehículos según corresponda.

5.8.3 Visualización.

La visualización en una simulación es una ayuda importante para comprender la realidad que se está modelando. En este modelo la visualización es importante por la gran cantidad de información que se genera cuando los vehículos se están moviendo por la ciudad. Por lo tanto se incorporó en el simulador la visualización del movimiento de los vehículos. El desarrollo de la visualización presentó un desafío en sí mismo. El simulador tuvo dos etapas en la visualización, en la primer etapa se utilizaba texto, en la segunda se evolucionó a una interfaz gráfica basada en el mapa de Montevideo.

En la etapa inicial la visualización presentaba diagramas de espacio tiempo tal como el de la Figura 5-2 . En la Figura 5-9 se muestra un ejemplo. Cada línea muestra el estado de una misma calle, tal como si se tomara una fotografía aérea de la calle. Los puntos son celdas que están vacías y los números son las velocidades de los vehículos. En la figura se ven dos vehículos que van avanzando a medida que pasa el tiempo. En el artículo de Nagel y Schreckenberg [66] se utiliza este tipo de visualización.

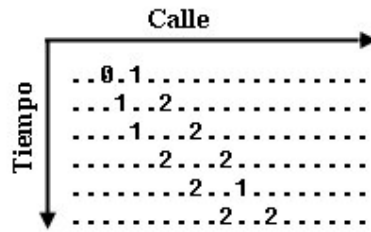


Figura 5-9. Diagrama de espacio tiempo.

Esta visualización es muy útil si se modela una sola calle, pero en una ciudad con muchas calles, es muy difícil poder ver el movimiento de los vehículos. Por lo tanto se evaluaron las interfaces de dos productos de simulación de tráfico. La interfaz que presenta Transims consiste esencialmente en representar los vehículos como puntos y los ejes de las calles [71]. Por otro lado, Mobility presenta una visión en tres dimensiones de las calles [78]. Se seleccionó la interfaz tipo Transims, ya que cumple con el objetivo de permitir ver adecuadamente la evolución de los vehículos y no es tan complicado de implementar como la de Mobility.

Para construir la interfaz gráfica se utilizó una biblioteca llamada CDX que es un kit de desarrollo de video juegos [79]. CDX es un conjunto de clases desarrollado como código abierto, que facilitan el uso de DirectX que es la plataforma de desarrollo de juegos de Microsoft [80]. Esta biblioteca facilitó la programación del componente de visualización. En la Figura 5-10 se puede ver la interfaz gráfica del simulador.

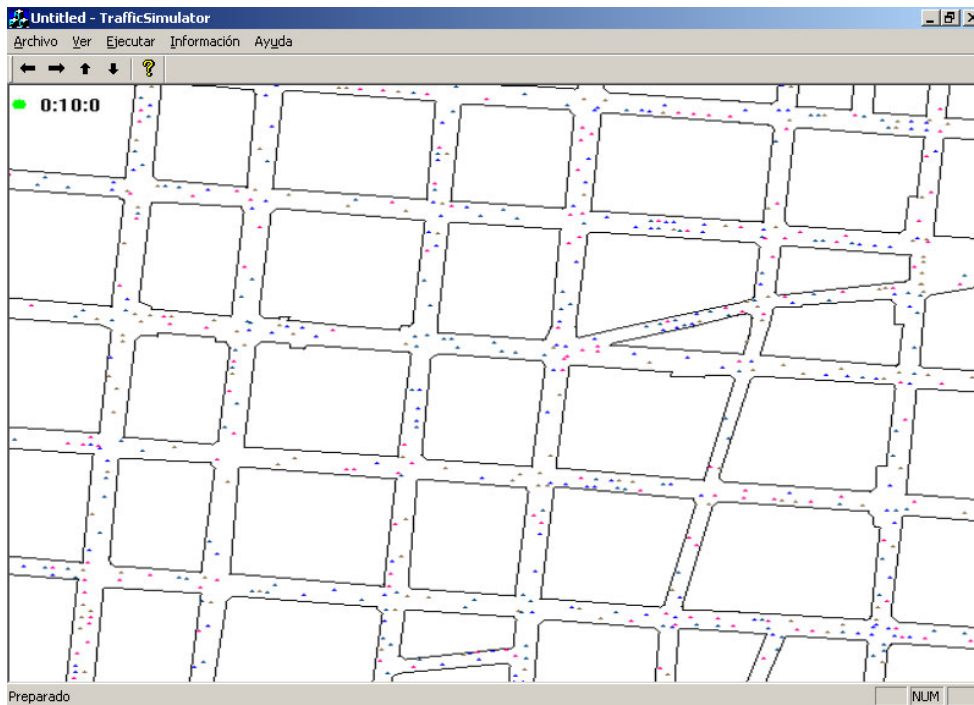


Figura 5-10. Interfaz gráfica del simulador.

Los puntos de colores son vehículos que se desplazan por las calles y las líneas indican las manzanas de la ciudad. En el borde superior izquierdo puede verse el tiempo transcurrido de la simulación y un círculo verde que indica que se alcanzó el estado estable. Eventualmente el círculo es rojo si no se alcanzó el estado estable. En la Figura 5-11 se pueden ver las celdas que forman las calles como puntos verdes.

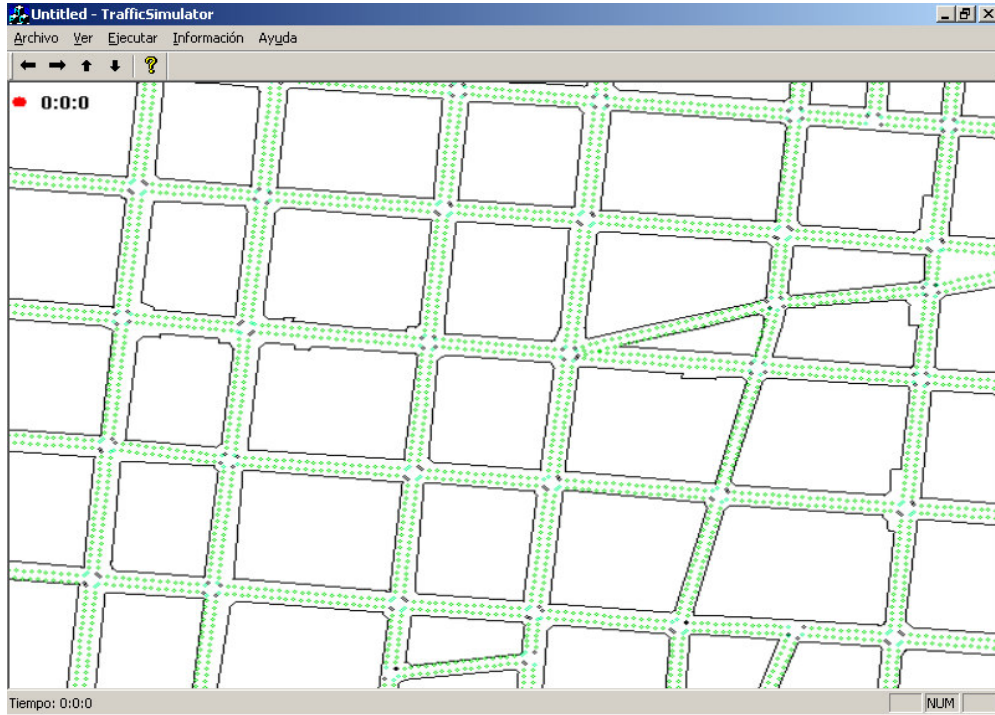


Figura 5-11. Celdas de las calles.

En la Figura 5-12 se las celdas marcadas por los círculos azules son las celdas que utilizan los sensores para tomar las mediciones.

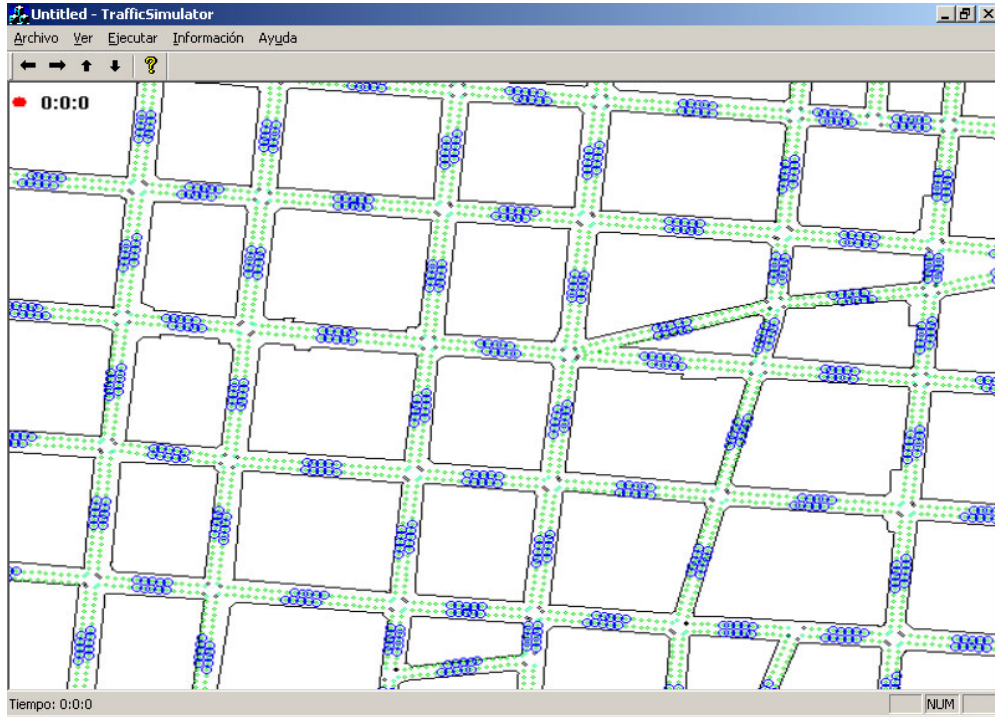


Figura 5-12. Zonas evaluadas por los sensores.

La Figura 5-13 muestra las celdas donde actúan los controladores, los círculos rojos indican controladores eliminadores y los negros controladores que insertan vehículos.



Figura 5-13. Controladores.

Finalmente, para visualizar gráficamente la evolución de los valores de densidad y flujo promedios de la ciudad se integró un control que genera gráficas [81]. En la Figura 5-14 se muestra en la gráfica de la izquierda la evolución de la densidad en el tiempo y en la gráfica de la derecha la evolución del flujo en el tiempo.

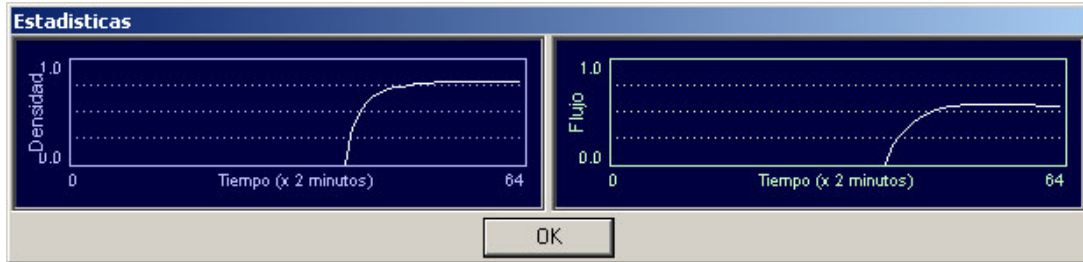


Figura 5-14. Evolución de la densidad y el flujo en función del tiempo.

5.8.4 Implementación.

La implementación del simulador fue desarrollada completamente en el lenguaje C++ utilizando como herramienta de desarrollo Visual Studio de Microsoft. El proyecto tiene aproximadamente 89 clases y más de 13.000 líneas de código. Se evaluaron algunas bibliotecas específicas para el trabajo con autómatas celulares, pero finalmente fueron descartadas porque el espacio del autómata, que en este caso está dado por el grafo de calles, no se pudo determinar si utilizando estas bibliotecas sería posible modelarlo.

5.8.4.1 Diseño de clases.

En la Figura 5-15 se ve un diagrama de Clases [63] del módulo encargado de realizar la Simulación de Tráfico. Este diagrama no tiene todas las clases del sistema, para preservar la claridad. En él se pueden ver solo las principales clases C++. Faltan muchas otras clases que implementan funcionalidades menores o clases que implementan el entorno gráfico de la aplicación.

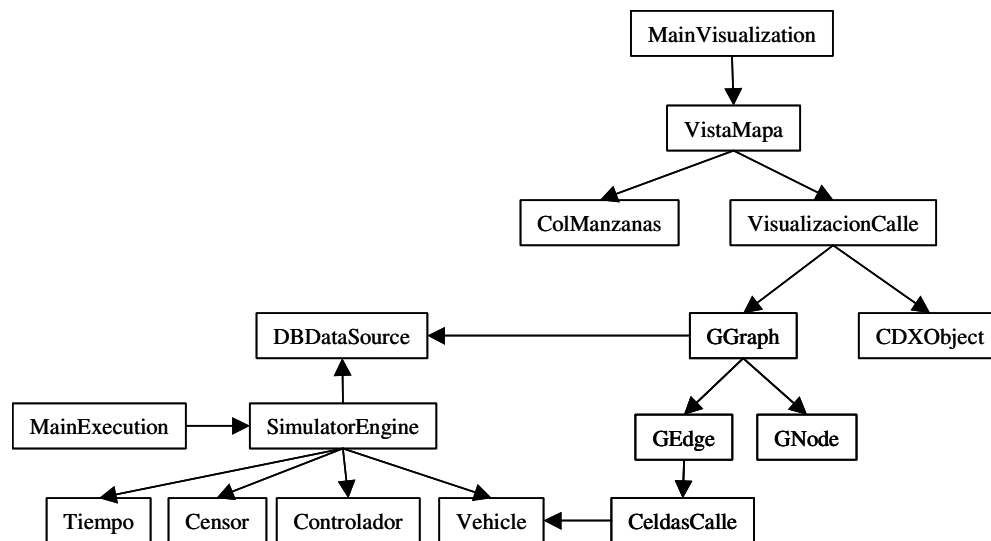


Figura 5-15. Diseño de clases.

Descripción de las clases:

GGraph – Implementa un grafo con las calles de la ciudad.

GEdge – Aristas del grafo. Las aristas del grafo se corresponden a las calles de la ciudad.

CeldasCalle – Implementa las celdas por las que se mueven los vehículos.

GNode – Nodo del grafo. El nodo se corresponde con las intersecciones de las calles de la ciudad.

DBDataSource – Esta clase se utiliza para implementar todos los accesos a la Base de Datos. La utiliza el grafo para cargar los datos de la ciudad y luego el motor del simulador para salvar los datos generados.

SimulatorEngine – Motor de la simulación. Es el encargado de manejar todo el comportamiento de la simulación, el movimiento de los vehículos, activar los controladores y sensores y actualizar el tiempo.

Tiempo – Modela un contador de tiempo.

Censor – Es la clase encargada de tomar las diferentes medidas que se utilizan durante la simulación.

Controlador – Implementa los controladores encargados de insertar o eliminar vehículos.

Vehículo – Modela a los vehículos que circulan por la ciudad.

MainExecution – Controla las acciones que son ejecutadas desde la interfaz gráfica del programa.

MainVisualization – Se encarga de la actualización de la vista del mapa de la ciudad.

VistaMapa – Se ocupa de mantener la vista del mapa que el usuario tiene, por ejemplo si el usuario realiza operaciones de paneo en esta clase queda registrada la información necesaria para que se mantenga esa vista.

ColManzanas – Mantiene información sobre las manzanas de la ciudad, para que puedan verse en el mapa del simulador.

VisualizaciónCalle – Se encarga de la visualización de una calle y de los vehículos que se encuentran en ella.

CDXObject – Biblioteca de objetos llamada CDX para la implementación de juegos. Con esta biblioteca se implemento la interfaz gráfica del mapa.

5.8.4.2 Parámetros configurables.

Algunos parámetros utilizados por el simulador de tráfico son configurados desde un archivo de inicio externo al programa. A continuación se listan estos parámetros.

Parámetro	Descripción
log	1 o 0 según se quiera la traza de la ejecución del programa.
logpath	Ubicación del archivo de la traza.
dsn	Nombre del Data Source para el acceso a la Base de Datos
maxspeed	Velocidad máxima (en celdas) a la que pueden ir los vehículos.
slowprob	Probabilidad de que un vehículo disminuya la velocidad.
cellengt	Largo de la celda.
loadboundingbox	Extensión del mapa que delimita la zona de la ciudad que va a ser simulada.
noexitroad	Lista de calles que quedan como callejones en el mapa.
lengthdivision	El largo de la calle se divide entre este parámetro para saber cuantos vehículos se recuerdan para

	medir la velocidad prevaleciente de la calle.
updatetimewhennoncars	Si se cumple este tiempo y no paso ningún vehículo por la calle, se toma una medición como si pasara un vehículo con velocidad máxima.
VelocidadCongestionRoja	Se indica que valor de velocidad de circulación se corresponde con el nivel de máxima congestión. Esto se utiliza para ver las calles congestionadas en color rojo, en Viajero Informado.
VelocidadCongestionAmarilla	Este parámetro es similar al anterior pero indica que valor de velocidad de circulación se corresponde con el nivel de congestión media.

A continuación se lista el archivo de configuración utilizado actualmente por el simulador de tráfico.

```
# Archivo de configuración de parámetros

[Log]
log=1
logpath=E:\desa\codigoc\TrafficSimulator\Release\log.txt

[DataBase]
dsn=atispj

[SimulatorVehicleConfig]
maxspeed=4
slowprob=0.3

[SimulatorCells]
celllengt=5

[SimulatorEngine]
loadboundingbox=464731.500000 6137064.850078, 466698.718750
6139034.149922
noexitroad=19521 19553 19732 19748 19782 19908 20726 21355
22095 22375 22412 22467 22468

[SimulatorTimeSensor]
lengthdivision=3
updatetimewhennoncars=0:0:30

[SimulatorCongestion]
VelocidadCongestionRoja=0
VelocidadCongestionAmarilla=1
```

5.9 Herramientas de simulación evaluadas.

Antes de comenzar con el desarrollo del simulador se estudiaron algunas herramientas de simulación microscópica de tráfico. Estas herramientas se describen brevemente a continuación.

5.9.1.1 Mobility.

Mobility [78] es un juego de planificación y administración de la red vial de una ciudad que utiliza técnicas reales. El desarrollo es apoyado por la Universidad de Bauhaus en Alemania. La herramienta permite diseñar la red de calles y el sistema de transporte público de una ciudad, entre otros aspectos. La visualización es en tres dimensiones y es muy buena. Mobility se descartó como herramienta de simulación para este proyecto ya que no es posible ingresar automáticamente un mapa digital como el de Montevideo (solo se puede hacer manualmente). Tampoco es posible obtener los datos del tráfico generados por la simulación para ser utilizados fuera de Mobility, es decir que no pueden ser exportados a archivos, por ejemplo.

5.9.1.2 Transims.

Transims [69] es una herramienta de predicción del transporte desarrollada por Los Alamos National Laboratory en Estados Unidos. Incluye módulos de simulación microscópica, planificación de rutas y estimación de la emisión de gases. El software fue descartado porque no es de uso libre.

5.9.1.3 AIMSUN2

AIMSUN2 [82] es una herramienta muy completa de simulación microscópica, desarrollada por la Universidad Politécnica de Cataluña, en España. Esta herramienta también fue descartada porque no es de uso libre.

6 Experimentos con el sistema.

Se realizaron dos tipos de experimentos con Viajero Informado. Por un lado se hicieron pruebas de desempeño del sitio web para poder estimar cuales serían las condiciones en las que debe ser instalado para su ejecución en un ambiente de producción (es decir utilizado por usuarios reales). Por otro lado se realizaron experimentos para estudiar los algoritmos de planificación de rutas y ver en que casos es conveniente aplicar el algoritmo que toma en cuenta la información de tráfico y el que no lo hace.

6.1 Pruebas de desempeño.

La intención de estas pruebas es poder determinar como utiliza la aplicación los distintos recursos del sistema. De este modo, si la aplicación fuese a ponerse en producción, se podría tener una estimación inicial de las características que debería tener ese ambiente.

En las aplicaciones web es muy importante realizar pruebas de desempeño, ya que potencialmente un sitio web puede ser accedido por miles de usuarios simultáneamente. Esta situación puede llevar al colapso si el sistema no está diseñado para soportar ese volumen de carga.

6.1.1 Pruebas de desempeño.

Las pruebas de desempeño de un sitio web sirven para comprender como responde el sitio web y el entorno, bajo distintos niveles de acceso por usuarios. Las medidas que normalmente se toman para un sitio web son latencia, cantidad de requerimientos procesados y utilización [83], [84]. Para asistir en el estudio de estas medidas se utiliza programas llamados 'de stress' que envían requerimientos a un sitio web simulando como si muchos usuarios lo estuvieran accediendo. Para estas pruebas se utilizó Web Application Stress Tool [85].

Latencia.

La latencia o tiempo de respuesta se refiere a la demora que experimenta el usuario desde que envía un requerimiento al servidor y este es respondido. La medida que se utiliza para medir la latencia en estas pruebas es el *tiempo hasta el último byte*, que calcula el tiempo que transcurre desde el requerimiento del usuario hasta que se recibe el último byte de respuesta. Un sitio web ideal debería tener la menor latencia posible. Generalmente, la latencia aumenta linealmente cuando aumenta la cantidad de usuarios, hasta determinado punto en que aumenta en forma exponencial [83].

Para comprender mejor el desempeño de la latencia, se la divide en dos tipos, latencia de la aplicación y latencia de la red. La latencia de la aplicación se refiere al tiempo que demora la aplicación en el procesamiento y latencia de red se refiere al tiempo que demoran los datos en viajar por las redes. En la Figura 6-1 se puede ver un

esquema del procesamiento general de una aplicación web y las distintas latencias de red (indicadas con R) y las de aplicación (indicadas con A). En este caso particular la latencia por el uso del Servicio Web del Servidor de Mapas está contemplada dentro del tiempo que demora la aplicación en el Servidor Web (A1 o A3).

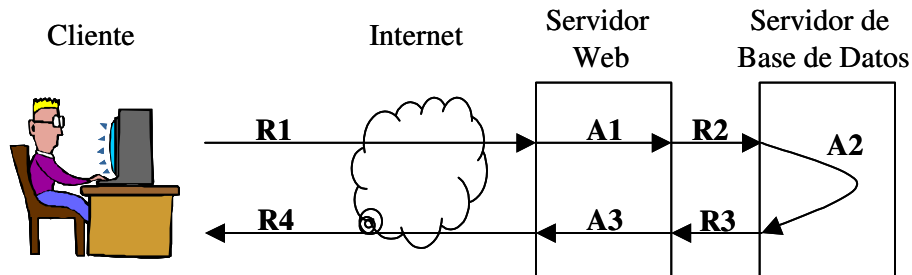


Figura 6-1. Latencia de la aplicación.

$$\text{LATENCIA TOTAL} = R1 + R2 + R3 + R4 + A1 + A2 + A3$$

Para las pruebas realizadas se cumple que las latencias de red son prácticamente cero porque el usuario cliente está utilizando la misma computadora que el servidor web y el servidor de Base de Datos, entonces no hay redes de por medio. Entonces en este caso la latencia estará dada principalmente por la latencia de la aplicación.

Requerimientos procesados.

Los requerimientos procesados se refieren a la cantidad de requerimientos del cliente procesados por unidad de tiempo. Los requerimientos pueden ser páginas web, imágenes o cualquier otro elemento que sea necesario que el usuario descarge. La unidad de medida es en este caso la cantidad de requerimientos por segundo. Generalmente el comportamiento de esta medida es que aumenta a medida que aumenta la cantidad de usuario, pero debido a que los recursos son limitados, tiende a disminuir luego de que cierta cantidad de usuarios se han alcanzado [83].

Utilización.

La utilización se refiere al nivel de uso de los distintos recursos del sistema como procesador o memoria RAM, entre otros. Generalmente lo que ocurre es que en todos los sistemas existen recursos que su disponibilidad se satura antes que la de los demás. Estos recursos configuran los cuellos de botella del sistema [83].

6.1.1.1 Prueba de stress.

La prueba consistió en ejecutar un test de stress [83] en el cual se simulaba como si tres usuarios requirieran la página inicial del Viajero Informado constantemente, durante cinco minutos. Es decir que cada uno de los tres usuarios requería la página inicial y cuando esta cargaba la volvía a requerir. Esta acción la hacia durante cinco minutos. Mientras se ejecutaba ese test se tomaron estadísticas de tres tipos, las propias estadísticas de la herramienta de stress, estadísticas de utilización de los recursos del sistema y estadísticas de acceso a la Base de Datos. A continuación se detallan los resultados obtenidos de esta prueba.

Selección de la página de prueba.

Desde el punto de vista del procesamiento que realizan las páginas web, en Viajero Informado existen tres tipos de páginas:

- La página inicial. Esta página tiene el trabajo de crear una nueva sesión del usuario, por lo tanto se insertan varios registros en la Base de Datos. Además muestra un mapa por lo que se realiza una invocación al Servidor de Mapas.
- Las páginas donde se muestran mapas. Por ejemplo la página donde se muestra la planificación de una ruta o el nivel de congestión de las calles. En estas páginas solo se actualizan algunos registros en la Base de Datos correspondientes a la sesión del usuario y se invoca al Servidor de Mapas. Estos tipos de páginas son las más frecuentes en el sitio.
- La página donde se realiza la planificación de rutas. Esta es la página donde se realiza más procesamiento. Esto se debe a la ejecución del algoritmo de planificación de rutas que realiza muchos accesos a la Base de Datos. En esta página no se muestran mapas.

Como la página inicial de Viajero Informado es una página intermedia en cuanto al procesamiento que se realiza y en el acceso a la Base de Datos, fue seleccionada para realizar las pruebas.

Estadísticas de la Base de Datos.

PostgreSQL toma diferentes estadísticas sobre la utilización de la Base de Datos [39]. En este caso se tomaron estadísticas sobre la utilización de las tablas de la aplicación y luego se utilizo la vista *pg_stat_user_tables* para ver el resultado [39]. El resultado de estas estadísticas se enumera en la Tabla 6-1. Solo se listan las tablas que fueron accedidas.

Tabla	Lect. Sec.	T. Rec. Sec.	Indices	Tup. Rec.	Inser.	Act.
Numnumerador	0	0	645	645	0	327
Spparcat	365	4380	0	0	0	0
Spesion	72	18383	0	0	21	0
Sppardef	0	0	675	4646	0	0
Spparses	0	0	1603	372173	830	0
Knscripsts	32	64	0	0	0	0
Debug	0	0	304	0	265	0
Geometry _columns	78	736	0	0	0	0
Totales	547	23499	3227	377464	1116	327

Tabla 6-1. Accesos a la Base de Datos.

Significado de las columnas:

Lect. Sec. – Cantidad de búsquedas secuenciales.

T. Rec. Sec – Cantidad de registros recuperados en las búsquedas secuenciales.

Indices – Cantidad de búsquedas utilizando índices.

Tup. Rec. – Cantidad de registros recuperados utilizando los índices.

Inser. – Cantidad de registros insertados.

Act. – Cantidad de registros actualizados.

En la fila de Totales se suman los valores de las columnas.

Por algún motivo desconocido los accesos realizados por el Servidor de Mapas no dejan estadísticas sobre las tablas que tienen información geográfica. Esto puede deberse a que PostGIS utiliza índices distintos a los que utiliza PostgreSQL en forma nativa y seguramente estos índices no registran estadísticas. En la Tabla 6-1 también se ve que una de las tablas que fue utilizada es la tabla Geometry_columns, que es una tabla de PostGIS que registra metadatos sobre las tablas con información geográfica. Esta tabla fue accedida por el Servidor de Mapas cuando presentó el mapa en la página inicial y como no tiene información geográfica, solo metadatos, se registran estadísticas.

Lo que puede concluirse de estos resultados es que algunas consultas no hacen uso de los índices, aunque en la mayoría de los casos no presentan demasiada actividad. Excepto en la tabla *Spesion* que si bien no se hacen muchas recorridas, se recuperan muchos registros. Por otro lado el mecanismo que almacena información sobre la sesión del usuario utiliza demasiados accesos a la Base de Datos. Las tablas que utiliza el mecanismo de sesión son: *Spparcat*, *Spesion*, *Sppardef* y *Spparses*. Por otro lado, no se hacen demasiadas inserciones y actualizaciones de registros.

Resultados de la utilización de los recursos del sistema.

Los recursos del sistema fueron medidos utilizando el programa *perfmon.exe* que es instalado por Windows 2000. Con este programa se pueden tomar distintas medidas de los recursos del sistema. Las medidas tomadas fueron:

- Memoria RAM disponible. Es la memoria RAM que queda libre para poder ser utilizada. La memoria está medida en Mega bytes y la capacidad máxima del sistema, en este caso es de 128 MB.
- Largo de cola de disco promedio. Es el número de requerimientos de operaciones de lectura o escritura del disco duro que son encolados porque no pueden atenderse en el momento que son requeridos.
- Porcentaje de utilización del procesador. Es el porcentaje del tiempo que el procesador no esta ejecutando procesos que no hacen nada (Idle process). Esta medida puede verse como el porcentaje de tiempo que la CPU ejecuta trabajo útil.

Las medidas fueron tomadas cada 10 segundos. Para obtener más información sobre estas medidas se sugiere examinar las definiciones que se hacen en el programa *perfmon.exe*.

En la Tabla 6-2 se presentan los resultado mínimos, máximos y promedio que fueron obtenidos durante la ejecución de la prueba de stress.

Recurso	Mínimo	Máximo	Promedio
Memoria RAM disponible (en MB)	1	27	17,3
Largo de cola de disco promedio (*100)	0,005	0,778	0,346
% utilización del procesador	4,404	96,493	81,571

Tabla 6-2. Resultados de la utilización de los recursos del sistema.

En la Figura 6-2 se ve la evolución de estas medidas a lo largo de la prueba.

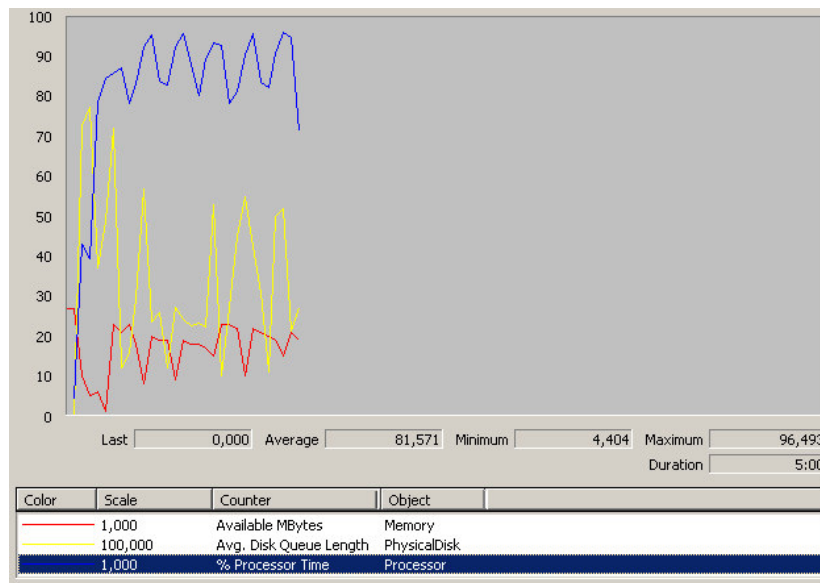


Figura 6-2. Evolución de la utilización de recursos del sistema.

Las características del PC utilizado son: 128 MB de memoria RAM, procesador Intel Pentium II, disco duro de 40 Gigabytes y sistema operativo Windows 2000. Un factor que influye en estas medidas también es que la aplicación de stress y la que toma las medidas ejecuta en el mismo PC que la aplicación web. Por lo tanto estas aplicaciones que también generan un consumo de recursos, están influyendo en las medidas.

Lo que demuestra esta prueba es que todos los recursos del sistema se vieron saturados por la aplicación, con tan solo 3 usuarios. Esto implica que la aplicación es altamente consumidora de recursos, principalmente de procesador y memoria. La Figura 6-2 lo que muestra es que en los picos de actividad (donde se utiliza más el procesador, en azul en la gráfica), la memoria disponible disminuye (en rojo en la gráfica) y los accesos al disco se ven demorados (en amarillo en la gráfica).

Resultados del stress web.

Los resultados del stress de la aplicación se muestran en la Tabla 6-3.

Medida	Valor
Tiempo total de ejecución	00:05:00
Requerimientos por segundo	0.06
Cantidad de usuarios	3
Errores	0
Cantidad de hits	18
THPB promedio	41784.67
THUB promedio	43870.11

Tabla 6-3. Resultados de la prueba de stress.

La explicación de las medidas es la siguiente [85]:

- Tiempo total de ejecución. Es el tiempo que duró la prueba.
- Requerimientos por segundo. Es el promedio de requerimientos por segundo durante la prueba.
- Cantidad de usuarios. Cantidad de usuarios concurrentes que requerían páginas web.
- Errores. Indica cuantas solicitudes fallaron.
- Cantidad de hits. Cantidad total de páginas web que fueron solicitadas.
- THPB. Tiempo hasta el primer byte, es decir el tiempo que demora en llegar el primer byte de respuesta en promedio. La medida es en milisegundos.
- THUB. Tiempo hasta el último byte, es el tiempo que demora en llegar el último byte de respuesta en promedio. La medida es en milisegundos.

Los resultados de las pruebas indican que la latencia (THUB) es excesivamente grande para ser una latencia de aplicación. También la cantidad de requerimientos por segundo es demasiado pequeña. Evidentemente la capacidad del sistema se vio excedida durante el test.

6.1.1.2 Desempeño de la planificación de rutas.

Para tener una medida del uso de los recursos durante la ejecución de una planificación de ruta, se realizó una prueba en la que se ingreso en Viajero Informado y se realizó una planificación. El proceso consistió en medir el desempeño del sistema y la cantidad de operaciones en la Base de Datos del mismo modo en que en la prueba anterior, pero sin hacer el stress del sitio web. De este modo se pretende tener una idea de cómo es la utilización de los recursos durante la planificación de rutas. El stress no se realizó debido a que la capacidad del sistema se vería rápidamente saturada y no se lograría tener una noción verdadera de la carga que representa la planificación de rutas.

La prueba consistió en ingresar a la página inicial de Viajero Informado, ingresar un usuario y contraseña válido y realizar la planificación de una ruta desde la esquina de José Enrique Rodó y Juan Paullier hasta la esquina de Uruguay y Tacuarembó. La selección de las esquinas de la planificación se debe a que la ruta es medianamente corta, la solución tiene 20 tramos.

Todo el proceso llevo aproximadamente 3 minutos y 50 segundos.

Resultados de la utilización de los recursos del sistema.

En la Tabla 6-4 se pueden ver los resultados de la utilización de recursos durante la prueba.

Recurso	Mínimo	Máximo	Promedio
Memoria RAM disponible	4	32	9,667
Largo de cola de disco promedio (*100)	0,014	1,087	0,403
% utilización del procesador	3,103	94,294	52,721

Tabla 6-4. Resultados de la utilización de los recursos del sistema.

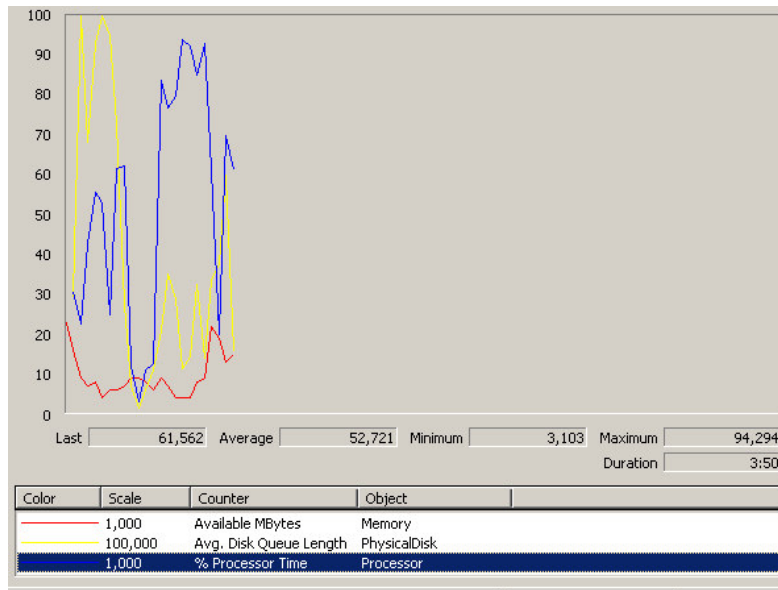


Figura 6-3. Evolución de la Utilización de recursos.

Los resultados de la Tabla 6-4 y las gráficas de la Figura 6-3 indican que si bien el proceso de planificar una ruta es muy intenso en el uso del procesador, no lo es tanto en el uso de memoria, ya que esta no desciende demasiado con respecto al resto de las actividades de la prueba. En la gráfica se puede ver claramente cuando el algoritmo de planificación es ejecutado debido al intenso uso del procesador (línea azul). Otro punto interesante de observar es el inicio de la prueba y el gran uso de operaciones de disco duro que se registran. Esto se puede deber al uso de la tecnología Java que carga las clases en memoria en el primer uso de la aplicación y luego las almacena en un caché, por eso luego no se registran usos tan intensivos de las operaciones de disco duro.

Utilización de la Base de Datos.

En la Tabla 6-5 se presentan los resultados de la utilización de la Base de Datos.

Tabla	Lect. Sec.	T. Rec. Sec.	Indices	Tup. Rec.	Inser.	Act.
numnumerador	0	0	110	110	0	55
Spparcats	79	948	0	0	0	0
Spsesion	16	4287	0	0	1	1
Sppardef	0	0	67	469	0	0
Spparses	0	0	209	48228	43	3
Knscripsts	6	12	0	0	0	0
Webusr	6	222	0	0	0	0
Debug	0	0	53	0	53	0
Laynombrecalles	0	0	8	10	0	0
Geometry_columns	13	120	0	0	0	0
Ruta	0	0	Sin dato	Sin dato	1	0
Rutatramos	0	0	1	0	20	0
GeoEjes2pdatos	167	3841140	0	0	0	0
GeoEjes	0	0	20	20	0	0
Ejecucionruteo	7	740	0	0	1	2

Tabla 6-5. Accesos a la Base de Datos.

En la utilización de la Base de Datos se destaca nuevamente la cantidad de accesos que se realiza para el manejo de la sesión del usuario. También se destaca la cantidad de accesos que se hacen en la tabla GeoEjes2pDatos. Esta tabla es la que tiene la información de las calles, que es la que utiliza el algoritmo de planificación de rutas si no utiliza información de tráfico, sino accede a otra tabla pero el comportamiento es el mismo (en cuanto al acceso a la información). Es razonable este resultado si se tiene en cuenta el modo en que fue desarrollado la carga del grafo de calles, que por cada nodo que explora recupera los nodos adyacentes. Sin dudas este es un punto donde la aplicación debe mejorar para disminuir la cantidad de accesos.

6.1.2 Arquitectura sugerida.

En esta Sección se dan las pautas para determinar cual debería ser una arquitectura ideal para poner a Viajero Informado en producción. Si bien no se menciona ninguna arquitectura en concreto, se dan las pautas generales que deben cumplir basándose en las pruebas descritas en las secciones anteriores.

La arquitectura sugerida para poner en producción a Viajero Informado es un servidor que tenga potenciado principalmente el procesador y la memoria RAM que son los recursos más consumidos por la aplicación. En cuanto al disco duro es recomendable que también haya suficiente espacio disponible, principalmente si la Base de Datos va a estar instalada en ese mismo servidor. Aunque, por la tecnología utilizada en el

acceso a la Base de Datos (ODBC), esta podría estar instalada en un servidor distinto del Servidor Web y simplemente cambiando la configuración del acceso a los datos (ODBC) Viajero Informado funcionaría sin necesidad de describir la aplicación. Así mismo, el Servidor de Mapas puede estar instalado en otro servidor distinto que el de Viajero Informado. Simplemente, cambiando uno de los parámetros de Viajero Informado la aplicación funcionaría correctamente sin necesidad de describirla.

En cuanto al Servidor de Mapas, es un potencial consumidor de disco duro ya que las imágenes que genera con los mapas son almacenadas en el disco duro. Si se tiene en cuenta que la mayoría de las páginas del sitio generan mapas, entonces el consumo de espacio en disco puede ser importante. Cabe destacar que las imágenes son utilizadas una única vez, es decir que luego de que son generadas y pasa un tiempo prudencial en el que se asegura que la imagen fue descargada por el usuario, entonces esta puede ser eliminada del disco.

Otro factor que se debe tener en cuenta es el ancho de banda que debe tener la red de comunicación para que la aplicación presente poca latencia en la comunicación con el usuario. Los tamaños de las páginas de Viajero Informado son muy similares entre sí ya que todas las páginas son prácticamente iguales, en ese sentido. El tamaño de la página de inicio es el siguiente:

- Código HTML de la página: 27 KB.
- Imágenes: 18.5 KB.
- Imagen del mapa: 20 KB.

El tamaño del mapa dependen de la cantidad de elementos geográficos que se muestran en él. Por ejemplo si se hace un zoom acercando muy grande y se muestran pocos elementos el tamaño será menor que si se muestra toda la ciudad.

Si por ejemplo se espera que el sitio, en un pico de actividad, sea utilizado por 10 usuarios y que cada uno descargue 5 páginas, entonces se transferirán aproximadamente:

$$10 \text{ (usuarios)} \times 5 \text{ (páginas)} \times 65.5 \text{ KB (por cada página)} = 3275 \text{ KB}$$

Es decir que se transferirán durante una actividad pico del sistema 3275 KB, por lo que la conexión que tenga el servidor deberá soportar esta transferencia de datos para atender esa demanda.

6.1.3 Mejoras al desempeño.

Los factores como latencia y requerimientos procesados dependen en su mayoría de la potencia del servidor en que esté instalada la aplicación. Si se quiere mejorar el desempeño de estos factores potenciando el servidor, se pueden lograr mejoras. Sin embargo la utilización de los recursos depende principalmente de las acciones que realiza la aplicación. En este sentido las pruebas indican que la aplicación tiene algunos puntos en los que debe mejorarse para optimizar el desempeño. Estos puntos son:

- Manejo de la sesión del usuario. El problema que tiene el manejo de la sesión es que por cada parámetro que recupera de la Base de Datos realiza una consulta. Esto podría optimizarse diseñando nuevamente el mecanismo para que todos los parámetros sean recuperados con un solo acceso a la Base de Datos.
- En los algoritmos de planificación de rutas, la recuperación de la información de la ciudad realiza demasiadas consultas a la Base de Datos. Esto podría hacerse de otro modo, por ejemplo recuperando secciones grandes del mapa.
- Los algoritmos de planificación de rutas no utilizan las estructuras de datos con las que se consiguen los mejores tiempos de recuperación de información. Para eso podría utilizarse Heaps [46] para almacenar los conjuntos de nodos y optimizar las operaciones de búsqueda.

6.2 Experimentos realizados con los algoritmos de planificación de rutas.

El objetivo de estos experimentos es cuantificar el beneficio de utilizar el algoritmo que tiene en cuenta las condiciones de tráfico contra el algoritmo que utiliza únicamente información de la distancia. Si consideramos que las personas cuando planifican una ruta, intuitivamente piensan siempre en el camino que implique menor distancia, se puede considerar esta prueba como una evaluación entre la forma humana de planificar una ruta y la forma en que lo haría un sistema que tiene información compleja y dinámica.

Para esto se realizó un experimento que consiste en utilizar los dos métodos de planificación de rutas en una misma zona de la ciudad, primero con poca congestión y luego con mucha congestión. Se compara la diferencia en el distancia recorrida y en el tiempo que se demora en llegar al destino.

En la Figura 6-4 se muestra el congestionamiento de las dos zonas de pruebas. En la zona de la izquierda hay menos congestión que en la zona de la derecha.

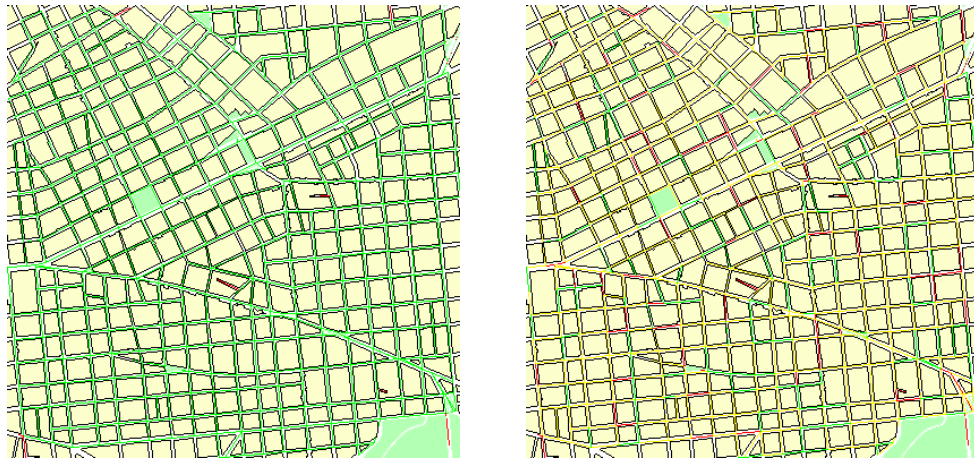


Figura 6-4. Congestionamiento de las zonas de prueba.

El experimento consiste en determinar un conjunto de quince orígenes y destinos dentro de la zona. Luego se utilizaron los dos algoritmos de planificación para realizar los recorridos con los distintos niveles de congestión. Utilizando otro programa desarrollado, se obtiene la distancia total recorrida y el tiempo que se demora en hacer el recorrido según las condiciones del tráfico. Los resultados se muestran en la Tabla 6-6 y corresponden a la suma de la distancia y el tiempo de las distintas rutas. La distancia esta medida en metros y el tiempo en segundos.

		Poca congestión	Mucha congestión
Ruteo Geográfico	Tiempo (seg)	1275	7887
	Distancia (mts)	22883	22883
Ruteo por Tráfico	Tiempo (seg)	1272	5305
	Distancia (mts)	26976	31416

Tabla 6-6. Resultados de las pruebas.

Congestión	Poca	Mucha
Tiempo (%)	0.3	33
Distancia (%)	15	27

Tabla 6-7. Comparación porcentual de los resultados.

En la Tabla 6-7 se puede ver la comparación porcentual de los resultados de la Tabla 6-6. Para obtener la diferencia porcentual del tiempo y distancias recorridas se realizó el siguiente cálculo:

Diferencia de distancia: $100 - (22883 \cdot 100 / 26976) = 15 \%$

Diferencia de tiempo: $100 - (1271.6 \cdot 100 / 1274.87) = 0.3 \%$

Diferencia distancia: $100 - (22883 \cdot 100 / 31416) = 27 \%$

Diferencia tiempo: $100 - (5304.58 \cdot 100 / 7887.27) = 33 \%$

Lo que indican los resultados de la Tabla 6-7 es que en condiciones de poca congestión el algoritmo que rutea utilizando la distancia es mejor ya que el que el ruteo utilizando la información de tráfico alargó las rutas un 15% pero solo disminuyó 0.3% el tiempo de llegada. Sin embargo en las situaciones con mucha congestión el resultado no es el mismo. En este caso el algoritmo que utiliza la información del tráfico aumentó 27% la distancia recorrida pero disminuyó el tiempo del viaje en 33%. Este resultado muestra que el algoritmo que utiliza la información de tráfico es una valiosa herramienta para que los conductores puedan disminuir el tiempo en sus viajes.

Los detalles de los experimentos y los resultados pueden encontrarse en el Apéndice C.

7 Conclusiones y Trabajo Futuro.

En este Capítulo se describen los aportes realizados por el trabajo, las conclusiones abordadas y el trabajo que queda pendiente para realizarse en el futuro.

7.1 Aporte de este trabajo.

Uno de los aportes más relevantes que se realizan en este trabajo es acercar al Uruguay el área de los Sistemas de Transporte Inteligentes y los Sistemas Avanzados de Información a Viajeros, que pueden ser un respuesta a los graves problemas que tiene el país en materia de seguridad y eficiencia en el tráfico.

En el momento en que se comenzó este trabajo el desarrollo de los SAIV estaba en pleno auge. Principalmente se impulsaba los sistemas desde un punto de vista comercial, para que fueran desarrollados o para que se mejoraran los que ya existían. A pesar de ese impulso y del tiempo transcurrido, en la practica se espera más funcionalidades de los sistemas que están actualmente funcionando. Este trabajo aporta el desarrollo de un prototipo y una arquitectura sobre la que se pueden desarrollar nuevos SAIV.

Por otro lado también se desarrolló un simulador de tráfico. Los simuladores de tráfico son herramientas imprescindibles en la planificación y predicción del tráfico. El simulador está adaptado a la ciudad de Montevideo, además de utilizar los mapas de la ciudad, también utiliza las restricciones de velocidad que se aplican.

Si bien el sistema, ya sea el simulador o Viajero Informado utilizan mapas de la ciudad de Montevideo, el desarrollo fue concebido para que los sistemas pudieran funcionar con cualquier ciudad de la que se pueda proveer los mapas digitales correspondientes. En este sentido las herramientas desarrolladas pueden adaptarse a nuevas ciudades, simplemente cargando los mapas en la Base de Datos y configurando las restricciones de velocidad en el archivo de configuración del simulador de tráfico.

El sistema desarrollado también resulta un interesante ejemplo de la integración de tecnologías. El prototipo integra componentes de: Sistemas de Información Geográfica y Servidores de Mapas, mapas digitales, Bases de Datos extendidas para el manejo de información geográfica, páginas web, servicios web, algoritmos de planificación de rutas y un simulador de tráfico.

Finalmente realizando pruebas con el sistema desarrollado se obtienen dos resultados interesantes. Por un lado se confirma el resultado intuitivo de que si hay pocas congestión de vehículos, ir hacia el destino utilizando el camino que recorre menor distancia es lo mejor. Sin embargo si hay más tráfico lo mejor es ir hacia el destino utilizando caminos que eviten la congestión. Un punto interesante que queda pendiente de estudio es determinar cual es el nivel de congestión en el que debe utilizarse una planificación que tenga en cuenta o no el tráfico.

El otro resultado que se obtiene es una cuantificación del beneficio que se obtiene de utilizar el algoritmo de planificación adecuado al nivel de congestión de la ciudad. Estas cuantificaciones cambian dependiendo de cada ciudad, por lo que este resultado obtenido puede tomarse como una base para la ciudad de Montevideo.

7.2 Conclusiones.

A continuación se describen las conclusiones luego del desarrollo de este proyecto.

7.2.1 STI y SAIV.

Los Sistemas de Transporte Inteligentes ya son realidad en los países desarrollados que tienen los problemas más graves en el transporte. Si bien hay un gran avance en cuanto a la implantación de componentes específicos, aún queda un largo camino para que los países tengan el sistema implantado a nivel nacional.

Durante el transcurso de este trabajo se constató un cambio en el rumbo de la disciplina de los STI. Actualmente se está prestando un especial interés en la seguridad del sistema de transporte frente a catástrofes o atentados, luego de los atentados a las torres World Trade Center en Estados Unidos. Esta nueva dirección se puede constatar con más claridad en Estados Unidos.

Sobre los Sistemas Avanzados de Información a Viajeros se puede decir que no han evolucionado demasiado. Desde que se comenzó este trabajo, a principios del año 2001, hasta el momento en que se escribe este informe, no se ve una evolución en la información y servicios que ofrecen. Generalmente en los sitios web dependientes del gobierno se ofrece principalmente información en tiempo real, de distintas formas, vistas de cámaras, mapas, etc. Sin embargo no hay servicios especiales como planificación de rutas. Seguramente esto se debe a que esos servicios especiales, de gran valor para el usuario serán provistos por los sitios web de empresas privadas o enfocados a sitios web accedidos exclusivamente por productos de telemática. Ahora hay más disponibilidad de sitios privados que ofrecen información en tiempo real. Es muy probable que en los próximos años este tipo de servicios sean más comunes. Por otro lado los sitios web de información de tráfico y tránsito continúan separados y generalmente no hay sitios que ofrezcan indistintamente ambas opciones integradas.

La apuesta inicial de este proyecto de utilizar Internet como plataforma para difundir información y servicios de tráfico fue acertada y hoy ya se pueden ver algunos sistemas que han tomado este mismo camino. Por ejemplo, la empresa Mercedes-Benz utiliza esta plataforma para brindar información a sus clientes, en los vehículos lanzados en el año 2003 [18]. Los clientes acceden a un sitio web donde tienen toda la información y servicios de tráfico. El acceso se hace mediante Asistentes Personales Digitales y comunicación inalámbrica. La combinación de aplicaciones web y Asistentes Personales Digitales es una excelente plataforma para acceder información y aplicaciones móviles en tiempo real. Es muy probable que esta combinación sea muy exitosa ya que cubre las necesidades de ejecución de aplicaciones móviles de una manera muy conveniente para el usuario.

7.2.2 Viajero Informado.

Viajero Informado demuestra que desarrollar un SAIV es una tarea compleja. Esto es debido a que el sistema se basa en la integración de componentes complejos y muy distintos entre sí. Cada componente debe funcionar coordinadamente con el resto para que el SAIV pueda brindar los servicios que se propone. Para la arquitectura definida fue muy importante que todos los componentes del sistema pudieran interactuar con la Base de Datos. También fue importante que la interacción entre los componentes funcionales del sistema pudieran resolverse de manera adecuada. De hecho el Servidor de Mapas fue integrado al sistema de una manera que no fue planificada originalmente por sus desarrolladores, que es a través de un Servicio Web.

Otro tema a destacar, es la disponibilidad de herramientas adecuadas para desarrollar el SAIV. Incluso algunos de los componentes más importantes del sistema como ser la Base de Datos y el Servidor de Mapas son de distribución libre, lo que hace pensar que esta modalidad de desarrollo es capaz de crear herramientas complejas y de muy buena calidad.

En cuanto a las pruebas de estrés realizadas con Viajero Informado se puede decir que hay que hacer algunos ajustes antes de que pueda ser instalado en un ambiente de producción. Si bien los resultados de las pruebas indican que se consume muchos recursos del sistema hay que tener en cuenta que estas fueron realizadas en un computador que no tiene las características de un Servidor Web sino las de un computador personal. Seguramente, en un Servidor Web se registren un uso menos intenso del procesador y de la memoria.

7.2.3 Algoritmos de planificación de rutas.

El algoritmo utilizado para la planificación de rutas, A* que es un algoritmo de búsqueda genérico, se adapta naturalmente al problema de buscar rutas. Si bien la implementación realizada no necesita cargar todo el mapa de la ciudad para realizar las búsquedas, puede optimizarse más aún. Cuando se recupera el grafo de la Base de Datos, se ejecutan demasiadas consultas que recuperan pocos datos. Seguramente si se recuperan secciones más grandes del grafo, el desempeño mejore porque el grafo se reconstruiría con menor cantidad de consultas a la Base de Datos pero que retornan más información.

Otra opción posible para mejorar el desempeño puede ser implementar un servidor de información de calles. Este servidor mantiene en memoria el grafo con la información de las calles. Entonces cuando un algoritmo de planificación requiere la información sobre las calles, se la solicita a este servidor en lugar de ir a la Base de Datos a buscarla. Los beneficios de utilizar este mecanismo son: los algoritmos no tienen que preocuparse de recuperar la información de la Base de Datos y de procesarla para armar una estructura de grafo ya que el servidor le puede enviar el grafo construido. Por otro lado, se evita la competencia entre los algoritmos de planificación de rutas por el acceso a los datos en la Base de Datos, evitándose así bloqueos y demoras. Las desventajas del mecanismo es que es necesario ocupar recursos de procesador y memoria para mantener al servidor funcionando. También el servidor debe encargarse de mantener actualizado los datos que mantiene en memoria cuando se realiza alguna

actualización en la Base de Datos y eventualmente notificar de los cambios a las aplicaciones que solicitaron información.

7.2.4 Simulación de tráfico.

La simulación del tráfico mediante simulación microscópica permite explorar con gran detalle los orígenes de los problemas de tráfico. El modelo utilizado cumplió adecuadamente con el objetivo de proveer información de tráfico. La complejidad de modelar el comportamiento del tráfico de una ciudad de una manera simple y clara muestra que los autómatas celulares pueden tener una gran capacidad para modelar comportamientos complejos de manera muy natural.

De la repetida ejecución de la simulación de tráfico, se observaron algunos detalles que tienen que ver con el comportamiento de los controladores y los sensores. El intervalo de tiempo en que actúan los controladores es un aspecto sensible en la simulación. Los controladores se basan en las estadísticas que obtienen los sensores para determinar si deben actuar o no. Entonces si los controladores actúan durante intervalos de tiempo muy cortos, los sensores no tendrán suficiente tiempo de registrar los cambios realizados por los controladores. Esto genera una situación de gran inestabilidad. En el otro extremo si los controladores actúan en intervalos más largos la situación tiende a ser más estable. Pero si actúan en intervalos excesivamente largos puede ocurrir que alcanzar la estabilidad tome demasiado tiempo. En la implementación se determinó que los controladores actúen cada dos minutos de tiempo simulado. Es decir que cada dos minutos los controladores consultaban las estadísticas de los sensores y determinaban si es necesario insertar o eliminar vehículos. Esto parece un tiempo razonable ya que llevó a que la simulación se comportara en forma estable.

Algo similar ocurre con la actualización de la velocidad de los sensores. Cuando no pasa ningún vehículo el sensor registra el pasaje de un vehículo imaginario con velocidad máxima, para reflejar el aumento en la velocidad de conducción en esa calle. El registro del pasaje de ese vehículo imaginario ocurre pasado cierto tiempo desde el último pasaje de un vehículo real. Si ese tiempo es demasiado corto se puede correr el riesgo que la velocidad siempre sea sobreestimada por la gran cantidad de vehículos imaginarios tenidos en cuenta. Si el tiempo es demasiado corto, entonces la velocidad determinada en esa calle no reflejará adecuadamente la realidad. En la simulación se consideró un tiempo de 30 segundos de inactividad por la calle para que se registre un vehículo imaginario que circula a velocidad máxima.

7.3 Trabajo futuro.

En esta Sección se describen algunos temas que en este trabajo quedaron pendientes de investigar y sería bueno que fueran abordados en trabajos futuros.

7.3.1 Algoritmos.

Sería interesante desarrollar variaciones del algoritmo de ruteo para que tome en cuenta las otras variables que provee el simulador, como flujo, densidad y velocidad. Como estas variables describen de distinta manera la situación en cada calle, es probable que los resultados varíen.

También sería bueno probar algoritmos de ruteo que consideran la naturaleza no lineal del tráfico de vehículos.

Por otro lado, los resultados de las pruebas realizadas indican que no siempre es conveniente utilizar el mismo algoritmo de planificación. Esto es debido a que cuando hay poco tráfico, para disminuir el tiempo de llegada en 0.3% se debe aumentar el 15% de la distancia del recorrido. Como el tiempo de viaje que se disminuye es tan pequeño no conviene realizar ese desvío. Por lo tanto, es importante poder determinar con precisión en que condiciones de tráfico debe dejar de utilizarse un método de búsqueda y utilizarse el otro. De lo contrario el usuario percibiría el mal desempeño de la solución propuesta y el sistema perdería la credibilidad, que es un factor crítico en el éxito del sistema.

Otro tema importante sobre el que se debería investigar es como lograr que los algoritmos funcionen cuando la información es incompleta. El simulador simula el tráfico de todas las calles de la zona seleccionada, por lo que el algoritmo de planificación siempre encontrará información. Sin embargo esta situación es prácticamente imposible de aplicarse en la realidad ya que sería necesario de instalar un censor por cada calle de la ciudad, lo que demandaría un costo excesivamente elevado. Entonces en el caso que se contara con algunos sensores en la ciudad, la información podría ser utilizada directamente, junto con valores predeterminados para las calles de las que no se tenga información. Otro enfoque posible es utilizar esta información para que ingrese en un simulador que continuamente está generando información sobre toda la ciudad y estima los niveles de congestión de las zonas donde hay información disponible y eventualmente tiene valores por defecto o configurados manualmente para las zonas donde no tiene información.

7.3.2 Viajero Informado.

Uno de los temas más interesantes en los que se puede profundizar en el desarrollo de Viajero Informado, es en la personalización de las herramientas para el usuario. Es decir que el sitio web pueda tener en cuenta las características y preferencias del usuario cuando este utilice los servicios. Algunos servicios interesantes que puede ofrecer son:

- Recordar los destinos más comunes del usuario, por ejemplo su lugar de trabajo o su hogar.
- Recordar preferencias sobre las rutas, por ejemplo rutas favoritas o rutas que no le agradan. Esto impactaría también en el modo en que se realizan las planificaciones de rutas.
- Tener modos alternativos de llegar al usuario sin que este tenga que ingresar al sitio web. Por ejemplo, puede ser muy conveniente para el usuario que el sitio le envíe un correo electrónico indicándole cual es la mejor ruta para llegar a su hogar, cuando se acerca la hora de salida del trabajo.

- Determinar cual es el comportamiento, dentro del sitio web, de cada usuario individual y de la mayoría de los usuarios para poder darle mayor relevancia a las aplicaciones preferidas de los usuarios y prever a cuales aplicaciones asignarles más recursos del Servidor Web.

Otro tema fundamental es implementar distintos modos de operación en Viajero Informado. Esto quiere decir que el sitio web se pueda comportar de distintas maneras dependiendo de la etapa del viaje en que se encuentra el usuario. Si se encuentra en la etapa previa al viaje, entonces el usuario seguramente es más receptivo a la experimentación con el sistema y al uso de varias herramientas para comprender el estado del tráfico en el momento que desea realizar el viaje. Sin embargo, si el usuario está en la ruta y necesita indicaciones, el sitio debe tener en cuenta que el usuario está prestando atención a la ruta y debe presentar el modo de operación más sencillo posible.

Un tema también interesante es poder medir el nivel de adherencia a las planificaciones de rutas. Es decir, si el usuario sigue al pie de la letra la ruta que se le planifica o le realiza modificaciones. Este tema tiene la dificultad de que es muy difícil que el usuario quiera indicar cual es la ruta que tomó en una determinada ocasión ya que posiblemente no la recuerde o no tenga ganas de ingresar toda esa información en el sitio web. Para facilitar esta tarea sería ideal poder integrar alguna tecnología de posicionamiento como GPS que permitan detectar en todo momento cual es la posición del usuario en la ciudad y así verificar de manera más sencilla la adherencia.

Finalmente sería interesante implementar distintas versiones del sitio web para accederlo desde distintas plataformas, principalmente de plataformas móviles como ADP o incluso teléfonos celulares.

7.3.3 Simulación.

Conseguir algún tipo de información sobre el flujo verdadero del tráfico de vehículos en la ciudad de Montevideo y aplicarlo a la simulación, sería importante para obtener resultados más reales. Además se podría verificar si la simulación del tráfico de la ciudad puede alcanzar la estabilidad o no. En este trabajo se supuso que el tráfico es estable pero en realidad esto debería ser comprobado utilizando datos reales.

También sería interesante conseguir otras capas de información como localización de semáforos, información sobre cebras para el cruce de peatones o información sobre restricciones de velocidad. Esto no solo se aplicaría en la simulación, sino que también los algoritmos de planificación la pueden tener en cuenta para ajustar sus resultados.

Por otro lado, las reglas de movimiento del modelo son extremadamente simples. Si bien reflejan a grandes rasgos las características del tráfico, conducir un vehículo es una actividad extremadamente compleja. Las reglas pueden ser ampliadas para tener en cuenta distintos factores, por ejemplo distintos tipos de conductores con distintos niveles de prudencia, distintas modalidades de viaje, por ejemplo, apurado por llegar al destino o de paseo, etc.

La forma de modelar los cruces es un tema poco estudiado. Los modelos expuestos en este trabajo no son demasiado reales. El modelo de las rotondas tiene algunos inconvenientes para implementar reglas de preferencia, por ejemplo señales de pare o ceda el paso. En este modelo el vehículo que está en la rotonda necesariamente tiene que tener la preferencia para salir, sino se podría bloquear la rotonda porque los vehículos tienen preferencia para entrar. Es necesario crear nuevos modelos que permitan modelar reglas de circulación reales como la de la mano derecha (un vehículo que viene circulando por la derecha de otro tiene la preferencia de atravesar primero el cruce).

Sería interesante agregar incidentes en la simulación, como accidentes de tránsito u obras de construcción y poder determinar como impacta eso en la zona. También sería interesante que los incidentes sean tenidos en cuenta explícitamente por los algoritmos de planificación de rutas. Implícitamente los algoritmos lo tendrán en cuenta ya que un incidente generalmente incrementa la congestión en la zona y afecta los tiempos de viaje. Entonces el algoritmo debería advertir el incidente y estimar una zona de influencia para evitarla.

También sería interesante adaptar el simulador para que pueda ejecutar continuamente y cada cierto tiempo, ingrese la información generada, automáticamente en la Base de Datos. En la implementación actual el simulador debe detener la ejecución para habilitar esa opción. Con la implementación actual ese trabajo es muy sencillo ya que el desarrollo ya está hecho. Lo interesante sería poder estimar los niveles de carga que tendría el sistema en estas nuevas condiciones en cuanto a uso de los recursos y operaciones en la Base de Datos. Además se presentarían los problemas de consistencia en la Base de Datos mencionados en la Sección 4.4.4 y también se presentaría el problema en el que la información varía mientras una planificación de ruta se está llevando a cabo.

Bibliografía.

- [1] Lappin, J. “Advanced Traveler Information Service (ATIS): What do ATIS customers want?”. US Department of Transportation. January 2000. Reporte técnico disponible en Internet: http://www.its.dot.gov/itsweb/EDL_webpages/webpages/SearchPages/Alpha_Search.cfm. Número: 12284. Última visita: 30/7/2003.
- [2] Sussman, J. et al. “What have we learned about ITS?”. US Department of Transportation. December 2000. Reporte técnico disponible en Internet: http://www.its.dot.gov/itsweb/EDL_webpages/webpages/SearchPages/Abstract.cfm?docnumber=13316. Última visita: 30/7/2003.
- [3] Soolman, J. “Features of Traffic and Transit Internet Sites. US Department of Transportation”. February 2000. Reporte técnico disponible en Internet: http://www.its.dot.gov/itsweb/EDL_webpages/webpages/SearchPages/Abstract.cfm?docnumber=12263. Última visita: 30/7/2003.
- [4] Lappin, J. “Advanced Traveler Information Service (ATIS): Who are ATIS Customers?”. US Department of Transportation. January 2000. Reporte técnico disponible en Internet: http://www.its.dot.gov/itsweb/EDL_webpages/webpages/SearchPages/Alpha_Search.cfm. Número: 12285. Última visita: 30/7/2003.
- [5] El Uruguay en Cifras 2003. Instituto Nacional de Estadísticas. Página web, disponible en Internet: http://www.ine.gub.uy/biblioteca/uruguayencifras2003/uruguay_en_cifras2003.htm. Última visita: 26/8/2003.
- [6] *Intelligent Transportation Primer*. Institute of Transportation Engineers, ITS America and USDOT. Libro publicado por el Institute of Transportation Engineers. 2000. ISBN 0-935403-45-0.
- [7] Sitio web del Departamento de Transporte de Estados Unidos, sección Preguntas Frecuentes: <http://www.its.dot.gov/faqs.html>. Última visita: 14/8/2003.
- [8] Institute of Transportation Engineers. “A Toolbox for alleviating traffic congestion and enhancing mobility.” 1997. ISBN: 0-935403-12-4.
- [9] Li, I. Notas del curso “Traffic Management and Control” Universidad de Florida. Disponibles en Internet: <http://www.ce.ufl.edu/~ili/teaching/TTE6257/TTE6257.html>. Última visita: 15/8/2003.
- [10] Bishop, R. “Intelligent Vehicle Applications Worldwide”. IEEE Intelligent Systems. Enero/Febrero 2000. Páginas 79-81.

- [11] Tsugawa, S. "An introduction to Demo 2000: The cooperative driving scenario". IEEE Intelligent Systems. Julio/Agosto 2000. Páginas 78-79.
- [12] Iwasaki, Y. "The Current Status of ITS in Japan". Publicado en "Public Roads." Número 3, Volumen 61. Noviembre / Diciembre 1997. Disponible en Internet: <http://www.tfhr.gov/pubrds/pr97-12/p27.htm>. Ultima visita: 16/8/2002.
- [13] Blossville, J., and M. Parent. "The French Program La Route Automatisée". IEEE Intelligent Systems. Mayo/Junio 2000. Páginas 10-13.
- [14] Mitretek Systems. "Intelligent Transportation Systems Benefits and Cost. 2003 Update". Manual técnico, disponible en Internet: <http://www.benefitcost.its.dot.gov>. Ultima visita: 19/8/2003.
- [15] "Deploying the integrated metropolitan Intelligent Transportation Systems (ITS) infrastructure. FY 2002 Report". Reporte publicado en Internet disponible en: <http://itsdeployment2.ed.ornl.gov/its2002/metroresults.asp>. Ultima visita: 19/8/2003.
- [16] Telematic Research Group Inc. "DaimlerChrysler Telematics in Europe". Reporte publicado en Internet: <http://www.telematicsresearch.com>. Ultima visita: 22/8/2003.
- [17] Yilin Zhao. "Vehicle location and navigation systems." 1997. ARTech House. ISBN 0-89006-861-5.
- [18] Telematic Research Group Inc. "Saab Telematics in North America". Reporte publicado en Internet: <http://www.telematicsresearch.com>. Ultima visita: 22/8/2003.
- [19] Maclean, S. and D. Dailey. "The Use of Wireless Internet Service to Access Real-Time Transit Information". Publicado en World Congress, 2002. Disponible en Internet: http://www.its.washington.edu/its_pubs.html.
- [20] Shekhar, S. and D. Liu. "Genesis and Advanced Traveler Information Systems: Killer Applications for Mobile Computing". MOBIDATA. Volumen 1, número 1, Noviembre 1994. Disponible en Internet: <http://www.cs.rutgers.edu/~badri/journal/cover11.html>
- [21] Dillenburg, J. "Internet based traffic congestion information for the Gary-Chicago-Milwaukee Priority Corridor". Reporte técnico de la Universidad de Illinois. Disponible en Internet: <http://citeseer.nj.nec.com/67366.html>. Ultima visita: 30/7/2003.
- [22] Padmos, J. and D. Bernstein. "Personal Travel Assistants and the World Wide Web". The Transportation Research Board. 76th Annual Meeting. January 12-16, 1997.

- [23] Sitio web: Georgia Department of Transportation's Navigator: <http://www.georgianavigator.com/traffic/>. Ultima visita: 2/8/2001.
- [24] Sitio web: Athens Real Time traffic Map: <http://frida.transport.civil.ntua.gr/map/>. Ultima visita: 6/8/2001.
- [25] ITS America and U.S. Department of Transportation. "Closing the Data Gap: Guidelines for Quality Advanced Traveler Information System (ATIS) Data. Version 1.0". September 2000. Reporte técnico publicado Internet: http://www.itsdocs.fhwa.dot.gov/JPODOCS/REPT_MIS/13580.html. Ultima visita: 6/11/2003.
- [26] Picado, R. "Route Guidance". LEAP Intelligent Transportation Systems, Universidad de Berkeley. Página web: http://www.path.berkeley.edu/~leap/TTM/Route_Guidance/. Ultima visita: 19/2/2000.
- [27] Universidad de Princeton. "Pulsar System". Página web: <http://www.sor.princeton.edu/~dhh/PULSAR>. Ultima visita: 2/3/2000.
- [28] Universidad de Waterloo. "Dynamic Route Guidance System (DRGS)". Página web: <http://panda.uwaterloo.ca/~edward/projs/sd362>. Ultima visita: 19/2/2000.
- [29] Ben-Akiva, M. et al. "Development of a route guidance generation system for real-time application". Reporte técnico disponible en Internet: <http://its.mit.edu/projects/dta/dynamit.html>. Ultima visita: 19/2/2000.
- [30] Los Alamos National Laboratory. "Transims". Página web: <http://transims.tsasa.lanl.gov>. Ultima visita: 15/2/2000.
- [31] ITS America. "Choosing the Route to Traveler Information Systems Deployment". 1998. Publicado por ITS America. Disponible en Internet: [http://www.itsa.org/committe.nsf/0/458be804b5e125ae852566940055f81a/\\$FILE/aguide.pdf](http://www.itsa.org/committe.nsf/0/458be804b5e125ae852566940055f81a/$FILE/aguide.pdf).
- [32] Kolodko, J. and L. Vlacic. "Cooperative Autonomous Driving at the Intelligent Control Systems Laboratory". IEEE Intelligent Systems. Julio/Agosto 2003. Páginas 8 a 11.
- [33] Fletcher, L., et al. "Vision in and out of Vehicles". IEEE Intelligent Systems. Mayo/Junio 2003. Páginas 12 a 17.
- [34] Wu, C. et al. "An Advanced Traveler Information System with Emerging Network Technologies". A publicarse en Proceedings of the 6th Asian-Pacific Intelligent Transportation System Forum, October 2003.
- [35] Mensaje en ITS Forum. Autor: Kirk Luckscheiter. Fecha: 17 de Julio del 2003, a las 4:41 p.m. Título: "Re: How Can We Address the IVI "Chicken-and Egg Situation?". Disponible en Internet:

<http://www.nawgits.com/itsforum/nawg/index.cgi?rev=1599>. Mensaje número:1599.

- [36] Magney, P. "The Global State of Telematics". January 5, 2003. Telematic Research. Reporte publicado en Internet: <http://www.telematicresearch.com>.
- [37] Korth, H. y A. Silberschatz. "Fundamentos de bases de datos." 1993. Mc Graw Hill. ISBN 84-481-0079-4.
- [38] ESRI. "ESRI Shapefile Technical Description". Reporte técnico disponible en Internet: <http://www.esri.com>. Ultima visita: 2/3/2000.
- [39] Sitio web de PostgreSQL: <http://www.PostgreSQL.org>. Ultima visita 30/7/2003.
- [40] Sitio web de PostGIS: <http://postgis.refractory.net/>. Ultima visita 30/7/2003.
- [41] Sitio web de Open GIS Consortium: <http://www.opengis.org>. Ultima visita: 30/7/2003.
- [42] Giosa, D., L. Tansini. "Interoperabilidad en Sistemas de Información Geográfica". No publicado. Instituto de Computación. Universidad de la Republica Oriental del Uruguay. 1999.
- [43] Open GIS Consortium. "Simple Features for Specification for SQL". Estándar publicado por Open GIS Consortium. Disponible en Internet: <http://www.opengis.org/techno/specs/99-049.pdf>. Ultima visita: 30/7/2003.
- [44] Genexus. Sitio web de ARTech: <http://www.artech.com.uy>. Ultima visita: 25/8/2003.
- [45] Sitio web de Intendencia Municipal de Montevideo: <http://www.imm.gub.uy>. Ultima visita 30/7/2003.
- [46] Aho, A., J. Hopcroft y J. Ullman. "Estructuras de datos y algoritmos." 1988. Addison-Wesley. ISBN 201-64024-4.
- [47] Huang, Y., N. Jing and E. Rundensteiner. "Effective Graph Clustering for Path Queries in Digital Map Databases". Reporte técnico disponible en Internet: <ftp://ftp.eecs.umich.edu/people/rundenst/papers/INDEX.html>. Ultima visita: 19/2/2000.
- [48] Sitio web de MapServer: <http://MapServer.gis.umn.edu/>. Ultima visita: 30/7/2003.
- [49] Web Services. Sitio web de W3C: <http://www.w3.org/2002/ws/>. Ultima visita: 26/8/2003.
- [50] "HTTP - Hypertext Transfer Protocol." Página web publicada en el sitio de W3C: <http://www.w3.org/Protocols/>. Ultima visita: 26/8/2003.

- [51] "Naming and Addressing: URIs, URLs." Página web publicada en el sitio de W3C: <http://www.w3.org/Addressing/>. Ultima visita: 26/8/2003.
- [52] "Extensible Markup Language (XML)." Página web publicada en el sitio de W3C: <http://www.w3.org/XML/>. Ultima visita: 26/8/2003.
- [53] "HyperText Markup Language (HTML) Home Page." Página web publicada en el sitio de W3C: <http://www.w3.org/MarkUp/>. Ultima visita: 26/8/2003.
- [54] Lenguaje de programación Javascript. Puede encontrarse información en Internet: <http://www.javascript.com>. Ultima Visita: 3/9/2003.
- [55] Applets Java. Información disponible en la página web de Sun Microsystems: <http://java.sun.com>. Ultima Visita: 3/9/2003.
- [56] Greenwood, R. "jBox How to". Disponible en Internet: http://greenwoodmap.com/MapServer/jBox/jBox_docbook.html. Ultima Visita: 19/2/2003.
- [57] Esser, J. and M. Schreckenber. "Efficiency of Route Guidance Systems in Urban Road Networks". Reporte técnico de la Universität Duisburg-Essen. 1997. Disponible en Internet: <http://citeseer.nj.nec.com/177178.html>.
- [58] Russell, S. y P. Norving. "Inteligencia Artificial, un enfoque moderno." 1996. Prentice Hall. ISBN 968-880-682.
- [59] Cevallos, F. "Visual C++ 6. Aplicaciones para Win32." 2000. Alfaomega. ISBN 970-15-0446-1.
- [60] Biblioteca de clases desarrollada por M. Henri. Disponible en Internet: <http://www.markhenri.com/sw/index.html>. Ultima visita: 30/10/2002.
- [61] "CiniFile". Sitio web de Codeguru: <http://codeguru.earthweb.com/misc/cinifile.shtml>. Ultima visita: 26/8/2003.
- [62] "ShapeLib". Disponible en Internet: <http://gdal.velocet.ca/projects/shapelib/shapelib.html>. Ultimo visita: 30/8/2001.
- [63] Unified Modelling Lenguaje. Sitio web disponible :<http://www.uml.org>. Ultima visita: 6/11/2003.
- [64] "Resin". Sitio web de Caucho Technology: <http://www.caucho.com>. Ultima visita 30/7/2003.
- [65] N. Gartner et al. "Traffic Flow Theory. A state of the art report". Reporte especial publicado por la Federal Highway Administration. Disponible en Internet: <http://www.tfrc.gov/its/tft/tft.htm>. Ultima visita: 30/7/2003.

- [66] Nagel, K. and M. Schreckenberg. "A cellular automaton model for freeway traffic". *Journal de Physique I*. December 1992, p. 2221.
- [67] Klar, A., R. Kuhne and R. Wegener. "Mathematical Models for Vehicular Traffic". *Surv. Math. Ind.*, 6 (1996), p. 215. Disponible en Internet: <http://citeseer.nj.nec.com/klar96mathematical.html>. Ultima visita: 29/7/2003.
- [68] Chowdhury, D., L. Santen and A. Schadschneider. "Simulation of vehicular Traffic: A statistical physics perspective". *IEEE – Computing in simulation & engineering*. September – October 2000. Páginas 80 a 87.
- [69] Simon P. and K. Nagel. "Simplified cellular automaton model for city traffic". *Physical Review E*, volume 58, number 2. August 1998. Disponible en Internet: <http://www.sim.inf.ethz.ch/papers/>. Ultima visita: 29/7/2003.
- [70] Andriotti, G. and A. Bazzan. "An object Oriented microscopic traffic simulator". *InfoUY CLEI 2002, Congreso Latinoamericano de Informática*, Montevideo, Uruguay, Noviembre 2002.
- [71] Rickert, M. and K. Nagel. "Experiencias with a simplified microsimulation for the Dallas/Fort Worth area". *International Journal of Modern Physics C*. 1997. Disponible en Internet: <http://www.sim.inf.ethz.ch/papers/>. Ultima visita: 29/7/2003.
- [72] Chopard, B., P. Queloz and P. Luthi. "Traffic models of a 2D road network". *Proceedings of the 3rd CM users' Meeting*, Parma, Octubre 1995. Disponible en Internet: <http://cui.unige.ch/~chopard/Traffic/ca-models.html>. Ultima visita: 29/7/2003.
- [73] Dupuis, A. and B. Chopard. "Parallel simulation of traffic in Geneva using cellular automata". *Parallel and Distributed Computing Practices Journal*, 1(3):79-92, September 1998. Disponible en Internet: <http://cui.unige.ch/~dupuis/publications.html>. Ultima visita: 29/7/2003.
- [74] Esser, J. and M. Schreckenberg. "Efficiency of Route Guidance Systems in Urban Road Networks". *Reporte técnico de la Universität Duisburg-Essen*. 1997. Disponible en Internet: <http://citeseer.nj.nec.com/177178.html>. Ultima visita: 29/7/2003.
- [75] Rennard, J. "Introduction to Cellular Automata". Página web disponible en Internet : <http://www.rennard.org/alife>. Ultima visita: 22/01/2001.
- [76] Benjaafar, S., K. Dooley and W. Setyawan. "Cellular Automata for Traffic Flow Modeling". *Reporte técnico disponible en Internet*: <http://citeseer.nj.nec.com/380458.html>. Ultima visita: 4/8/2003.
- [77] Davies, R. and R. O'Keefe. "Simulation Modelling with Pascal." 1989. Prentice Hall International. ISBN:0-13-811571-0.
- [78] Sitio web de Mobility: <http://www.mobility-online.de>. Ultima visita 4/8/2003.

- [79] Sitio web de CDX Game Development Kit: <http://www.cdplib.com>. Última visita : 28/5/2003.
- [80] "DirectX". Sitio web de Microsoft: <http://www.microsoft.com>. Última visita : 28/8/2003.
- [81] Malburg, M. "Oscilloscope/StripChart Control". Página web disponible en Internet: <http://codeguru.earthweb.com/controls/oscope.shtml>.
- [82] Barcelo, J. "Microscopic Traffic Simulation: A tools for the análisis and assessment of ITS systems". Universitat Politècnica de Catalunya. Reporte técnico publicado en Internet: <http://www.tss-ben.com>. Última visita: 22/08/2002.
- [83] Ching, A., Wagner, A. "Understanding Performance Testing". Artículo publicado en Internet. Disponible en: <http://msdn.microsoft.com/library/en-us/dnduwon/html/d5dplyover.asp?frame=true>. Última visita: 3/11/2003.
- [84] Ching, A., Wagner, A. "Collecting Performance Data". Artículo publicado en Internet. Reporte técnico disponible en: <http://msdn.microsoft.com/library/en-us/dnduwon/html/d5collection.asp?frame=true>. Última visita: 3/11/2003.
- [85] Microsoft Web Application Stress. Herramienta de software: disponible gratuitamente en Internet: <http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/itsolutions/intranet/downloads/webstres.asp>. Última visita: 3/11/2003.

Apéndice A - Descripción de las Tablas de la Base de Datos.

A continuación se describen las tablas de la Base de Datos y los atributos que tiene cada una.

Tabla SPParCat.

Categoría de los parámetros utilizados en los metadatos de las capas de información geográfica.

Atributo	Descripción
SPParCatId	Identificador.
SPParCatDsc	Descripción.
SPParCatIniPrc	Procedimiento para iniciar la categoría.
SPParCatUsd	Si el parámetro es usado.

Tabla SPParDef.

Tabla con la definición de los parámetros.

Atributo	Descripción
SPParId	Identificador.
SPParCatId	Categoría.
SPParNme	Nombre.
SPParTyp	Tipo.
SPParDsc	Descripción.
SPParDefVal	Valor por defecto.
SPParUsd	Si es utilizado o no.

Tabla SPParSes.

En esta tabla se instancian los parámetros con los valores particulares de la sesión.

Atributo	Descripción
SPParId	Identificador del parámetro.
SPSesId	Identificador de la sesión.
SPParCatId	Categoría.
SPPSVal	Valor del parámetro.

Tabla SPSession.

Contiene información de la validez de la sesión y del usuario.

Atributo	Descripción
SPSesId	Identificación de sesión.
SPSesIni	Hora de inicio.
SPSesEnd	Hora de finalización.
WUUsrId	Identificación del usuario.

Tabla GEDLyr01.

Esta tabla contiene información sobre las capas geográficas. Es utilizada por Viajero Informado para solicitar al Servidor de Mapas, las capas que deben estar presentes en el mapa.

Atributo	Descripción
GEDLyrId	Identificador de la capa
GEDLyrShpTyp	Tipo de capa
GEDLyrDsc	Descripción
GEDLyrSrc	Información sobre el origen de los datos
GEDLyrDCre	Día de creación
GEDLyrTCre	Hora de creación
GEDLyrBBXin	Menor valor de eje X
GEDLyrBBXax	Mayor valor de eje X
GEDLyrBBYin	Menor valor de eje Y
GEDLyrBBYax	Mayor valor de eje Y
GEDLyrBBZin	Menor valor de eje Z
GEDLyrBBZax	Mayor valor de eje Z
GEDLstRecNum	Cantidad de registros de la capa

Tabla GEDDtaMng01.

Esta tabla contiene datos administrativos (metadatos) sobre las capas geográficas.

Atributo	Descripción
GEDDMId	Identificador de datos
GEDLyrId	Identificador de la capa geográfica
GEDDMSFNme	Nombre del Shape File de donde fue creada
GEDDMDteCre	Día de creación
GEDDMTmeCre	Hora de creación
GEDDMDsc	Descripción de la capa
GEDDMSrc	Fuente de los datos
GEDDMTbInme	Nombre de la tabla donde fueron almacenada la capa

Tabla GEDDMtMI.

Esta tabla contiene información administrativa sobre los atributos que tiene una tabla con información geográfica.

Atributo	Descripción
GEDDMId	Identificador del atributo
GEDDMAtNme	Nombre
GEDDMAtDsc	Descripción
GEDDMAtTyp	Tipo

Tabla KNScripts.

En esta tabla se almacenan JavaScripts que son necesarios para la interacción con el usuario.

Atributo	Descripción
KNScId	Identificador.
KNScDsc	Descripción.
KNScCde	Código fuente del Script.

Tabla WebUsr.

Información básica del usuario de Viajero Informado.

Atributo	Descripción
WUUsrId	Identificación del usuario.
WUName	Nombre.
WULogIn	Nombre para ingresar al sistema.
WUPwd	Contraseña.

Tabla WebUsrI.

Esta tabla permite configurar las capas de información geográfica que el usuario ve en el mapa. Utilizando esta información el módulo que genera solicitudes al servidor de mapas puede requerir solo las capas que el usuario desea ver.

Atributo	Descripción
WUUsrId	Identificación del usuario.
GEDLyrId	Identificado de una capa de información geográfica.

Tabla Debug.

Esta tabla fue utilizada para detectar errores mientras se desarrollaba la aplicación. En la tabla se deja información de la ejecución de los programas.

Atributo	Descripción
DebId	Identificador
DebMsg	Mensaje de depuración
DebTime	Hora en que se deja el mensaje

Tabla EjecucionRuteo.

Esta tabla almacena información sobre si el origen y el destino que el usuario selecciona para la planificación de la ruta son válidos.

Atributo	Descripción
ERId	Identificador.
ERCalleOrigen1	Primer nombre de la esquina de origen de la ruta.
ERCalleOrigen2	Segundo nombre.
ERCalleDest1	Primer nombre de la esquina de destino de la ruta.
ERCalleDest2	Segundo nombre.
ERPtoOrig	Indica si las calles de origen se interceptan.
ERPtoDest	Indica si las calles de destino se interceptan.
ERRutaId	Identificador de la ruta a generar.

Tabla LYNomCalle.

Tabla con los nombres de las calles de la capa de ejes.

Atributos	Descripción
Calle	Nombre de la calle.

Tabla SimLyrMedidas.

Tabla que almacena una capa de información generada a partir de los datos obtenidos por la simulación de tráfico.

Atributos	Descripción
SimCalleId	Identificador del eje.
SimCallDensidad1	Densidad de un sentido del eje
SimCallFlujo1	Flujo
SimCallTiempo1	Tiempo en atravesar el eje
SimCallVelocidad1	Velocidad prevaleciente
SimCallDensidad2	Densidad del otro sentido del eje
SimCallFlujo2	Flujo
SimCallTiempo2	Tiempo en atravesar el eje
SimCallVelocidad2	Velocidad prevaleciente
SimCallTipo	Doble via o simple via
SimCallSentido1	Sentido del eje
SimCallSentido2	Sentido del eje
SimCallCongestion	Nivel de congestión (rojo, amarillo o verde)

Tabla Numerador.

En esta tabla se almacenan los registros necesarios para generar los números de los identificadores necesarios.

Atributos	Descripción
NumId	Identificador.
NumDsc	Descripción.
NumUltNro	Ultimo número utilizado.
NumFchUltMod	Fecha de la ultima modificación.
NumHorUltMod	Hora de la ultima modificación.

Tabla Ruta.

En esta tabla se almacena la información sobre las rutas que deben ser planificadas.

Atributos	Descripción
RutaId	Identificador de la ruta
RutaOrigen	Origen
RutaDestino	Destino
RutaCntTram	Cantidad de tramos de la ruta

Tabla RutaTramos.

La tabla de RutaTramos almacena la información de cada uno de los tramos de la ruta que conforman una planificación de una ruta.

Atributos	Descripción
RutaId	Identificador de ruta
RutaTramId	Identificador de tramo
RutatramEjId	Identificador de eje
RutaTramLne	Eje geográfico de la ruta

Tabla GeoEjes.

Información de las calles de la ciudad de Montevideo.

Atributos	Descripción
geoejid	Identificador del eje
length	Largo
nombre	Nombre de la calle
geoejlne	Línea geográfica con el eje
Otros	Otros atributos que se encuentran en el archivo original que no son utilizados

Tabla GeoEsq

Información de las esquinas de las calles.

Atributos	Descripción
GId	Identificador del eje
GeoEsqId	Identificador de la esquina
GeoEsqPnt	Información geográfica de la esquina

Tabla GeoEjes2PDatos.

Información de las calles de la ciudad de Montevideo. Esta tabla es similar a GeoEjes, pero la información geográfica corresponde a otro tipo que la de GeoEjes. Esto es así para que PostGIS pueda soportar consultas sobre los puntos extremos del eje, que en el formato con que fueron almacenados en GeoEjes, no es soportado.

Atributos	Descripción
GId	Identificador del eje
GeoEsqIdIni	Esquina de inicio
GeoEsqIdEnd	Esquina de fin
Largo	Largo de la calle
Nombre	Nombre de la calle
GeoEj2PLne	Información geográfica de la calle

Tabla Manzanas.

Información sobre las manzanas de la ciudad.

Atributos	Descripción
GId	Identificador de la manzana
GeomManzanas	Forma geográfica

Tabla Playas.

Información sobre las playas de la ciudad.

Atributos	Descripción
GId	Identificador
GeomPlayas	Forma geográfica

Tabla Plazas.

Información sobre las plazas de la ciudad.

Atributos	Descripción
GId	Identificador
GeoPlazas	Forma geográfica
Otros Atributos	Que no son utilizados

Tabla Sep_Vial.

Información sobre los canteros y separadores viales de las calles de la ciudad.

Atributos	Descripción
GId	Identificador
GeomSep_Vial	Forma geográfica

Apéndice B - Manejo de parámetros.

Los parámetros están organizados en categorías, algunas de las categorías que se manejan son: capas geográficas, configuración del servidor de mapas, configuración del mapa en la página web, parámetros generales, parámetros para el manejo de archivos Shapefiles y capas geográficas definidas por el usuario. Cada parámetro está descrito por una fila en la tablas que se listan a continuación. La columna Categoría y Descripción de Categoría indican cual es la categoría a la que el parámetro pertenece y qué significa esa categoría. La columna Parámetro describe el nombre del parámetro. El Valor Inicial es el valor que tiene el parámetro en primera instancia, si corresponde, ese valor puede ser cambiado por la aplicación. Finalmente la columna Descripción indica el significado del parámetro.

Parámetros de visualización.

Con estos parámetros se puede configurar el ancho y alto de los mapas de Viajero Informado.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
InternetView	Parámetros de visualización para Interne	ImgH	400	Height de la imagen
InternetView	Parámetros de visualización para Interne	ImgW	400	Width de la imagen

Parámetros de las capas geográficas.

Los parámetros de las capas geográficas configuran el comportamiento, la visualización y el nivel de personalización que puede tener una capa geográfica.

Capa de Ejes.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrEjes	Layer Ejes	IsLayer	Y	Para definir a la categoría como de una layer.
LyrEjes	Layer Ejes	IsQueryOnly	N	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrEjes	Layer Ejes	IsUsrLayer	N	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta activo.
LyrEjes	Layer Ejes	LegendName	Ejes de calles	Nombre de la capa en la leyenda.

LyrEjes	Layer Ejes	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrEjes	Layer Ejes	MSLyrName	ejes	Nombre de la capa para MapServer.
LyrEjes	Layer Ejes	Visible	N	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Capa de Manzanas.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrManzanas	Layer Manzanas	IsLayer	Y	Para definir a la categoría como de una layer.
LyrManzanas	Layer Manzanas	IsQueryOnly	N	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrManzanas	Layer Manzanas	IsUsrLayer	N	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta activo.
LyrManzanas	Layer Manzanas	LegendName	Manzanas	Nombre de la capa en la leyenda.
LyrManzanas	Layer Manzanas	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrManzanas	Layer Manzanas	MSLyrName	manzanas	Nombre de la capa para MapServer.
LyrManzanas	Layer Manzanas	Visible	Y	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Capa de Playas.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrPlayas	Layer de Playas	IsLayer	Y	Para definir a la categoría como de una layer.
LyrPlayas	Layer de Playas	IsQueryOnly	N	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrPlayas	Layer de Playas	IsUsrLayer	N	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta activo.

LyrPlayas	Layer de Playas	LegendName	Playas	Nombre de la capa en la leyenda.
LyrPlayas	Layer de Playas	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrPlayas	Layer de Playas	MSLyrName	playas	Nombre de la capa para MapServer.
LyrPlayas	Layer de Playas	Visible	Y	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Capa de Plazas.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrPlazas	Layer Plazas	IsLayer	Y	Para definir a la categoría como de una layer.
LyrPlazas	Layer Plazas	IsQueryOnly	N	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrPlazas	Layer Plazas	IsUsrLayer	N	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta activo.
LyrPlazas	Layer Plazas	LegendName	Plazas	Nombre de la capa en la leyenda.
LyrPlazas	Layer Plazas	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrPlazas	Layer Plazas	MSLyrName	plazas	Nombre de la capa para MapServer.
LyrPlazas	Layer Plazas	Visible	Y	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Capa de Planificación de Ruta.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrRuteo	Layer ruteo del usuario	ERId	0	Indica el Id de la ejecución del ruteo
LyrRuteo	Layer ruteo del usuario	IsLayer	Y	Para definir a la categoría como de una layer.
LyrRuteo	Layer ruteo del usuario	IsQueryOnly	Y	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrRuteo	Layer ruteo del usuario	IsUsrLayer	Y	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de

				usuario solo se muestran cuando el usuario esta activo.
LyrRuteo	Layer ruteo del usuario	LegendName	Ruteo	Nombre de la capa en la leyenda.
LyrRuteo	Layer ruteo del usuario	LyrActive	N	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrRuteo	Layer ruteo del usuario	MSLyrName	ruteo	Nombre de la capa para MapServer.
LyrRuteo	Layer ruteo del usuario	Visible	N	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Capa de Separadores Viales.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrSep_Vial	Layer separador vial	IsLayer	Y	Para definir a la categoría como de una layer.
LyrSep_Vial	Layer separador vial	IsQueryOnly	N	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrSep_Vial	Layer separador vial	IsUsrLayer	N	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta activo.
LyrSep_Vial	Layer separador vial	LegendName	Separadores viales	Nombre de la capa en la leyenda.
LyrSep_Vial	Layer separador vial	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrSep_Vial	Layer separador vial	MSLyrName	sep_vial	Nombre de la capa para MapServer.
LyrSep_Vial	Layer separador vial	Visible	Y	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Parámetros para el manejo del Servidor de Mapas.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
MapServerParam	Parametros de MapServer	ImgBaseUrl	http://localhos	Url base para las imágenes creadas

				por MS.
MapServerParam	Parametros de MapServer	MSURL	/cgi-bin/mapserv36/mapserv.exe	URL de la instalación de Map Server.
MapServerParam	Parametros de MapServer	MapFile	mapserv36\mapa\sesionmap.map	Path del .map, con la info de layers, templates, etc.
MapServerParam	Parametros de MapServer	OperationMode	itemnquery	Modo de operación con MS, browse, query, itemquery, etc...
MapServerParam	Parametros de MapServer	QueryFile	mapserv36\mapa\querymap.map	Nombre del archivo .map para utilizar en las queries
MapServerParam	Parametros de MapServer	ServerName	localhost	Nombre del servidor web en donde esta instalado MapServer.
MapServerParam	Parametros de MapServer	ServerPort	80	Puerto del servidor web donde esta instalado map server.
MapServerParam	Parametros de MapServer	TrafficMap	mapserv36\mapa\simulationmap.map	Map file para hacer mapas con información de trafico
MapServerParam	Parametros de MapServer	TrafficQuery	mapserv36\mapa\simulationquerymap.map	Map para hacer consultas con información de trafico.

Parámetros del sistema.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
SystemParams	Parametros del sistema	AppletURL	http://localhost/cgi-bin/jbox/	URL de donde se encuentran las applets. Se usa con jBox.

Capa de puntos definidos por el usuario.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
LyrUserPoint	Layer puntos definidos por el usuario	IsLayer	Y	Para definir a la categoría como de una layer.
LyrUserPoint	Layer puntos definidos por el usuario	IsQueryOnly	Y	Indica si la capa es solo para hacer consultas, nunca se activa entera. Ej. capa de ruteo. Si el valor es Y, no se

				activa en la llamada a MapServer, pero si en la lista de capas activas.
LyrUserPoint	Layer puntos definidos por el usuario	IsUsrLayer	Y	Y si la capa es de usuario. N si no. Las capas que no son de usuario pueden mostrarse siempre. Las capas de usuario solo se muestran cuando el usuario esta logeado.
LyrUserPoint	Layer puntos definidos por el usuario	LegendName	Puntos de interés	Nombre de la capa en la leyenda.
LyrUserPoint	Layer puntos definidos por el usuario	LyrActive	Y	'Y' si la capa esta activada por el usuario (debe mostrarse), 'N' si no (no debe mostrarse).
LyrUserPoint	Layer puntos definidos por el usuario	MSLyrName	Puntos	Nombre de la capa para MapServer.
LyrUserPoint	Layer puntos definidos por el usuario	Visible	N	'Y' si la capa es visible para el usuario, 'N' si no. Para tener el control desde el Back Office

Manejo de Shapefiles.

Categoría	Descripción Categoría	Parámetro	Valor inicial	Descripción
ShpFiles	Actualizacion de archivos shape	AddLine	C:\Inetpub\wwwroot\shp\prgm\asetlines.exe	Path y nombre del programa que agrega lineas al .dbf (asetlines.EXE)
ShpFiles	Actualizacion de archivos shape	AddPoint	C:\Inetpub\wwwroot\shp\prgm\asetpoints.exe	Path y nombre del programa que inserta puntos en el .dbf (asetpoints.EXE)
ShpFiles	Actualizacion de archivos shape	AddShp	C:\Inetpub\wwwroot\shp\prgm\shpadd.exe	Path y nombre del commando shpadd.exe
ShpFiles	Actualizacion de archivos shape	CMDLneSeparator	#	Separador de campos de la línea de comando de los programas de actualización de puntos y líneas
ShpFiles	Actualizacion de archivos shape	LineDBF	C:\Inetpub\wwwroot\shp\wulines.dbf	Path y nombre del .dbf de las líneas. No se usa.
ShpFiles	Actualizacion de archivos shape	LineShpFile	C:\Inetpub\wwwroot\shp\wulines.shp	Path y nombre del archivo .shp de líneas

ShpFiles	Actualizacion de archivos shape	PointDBF	C:\Inetpub\wwwroot\shp\Wupoints.dbf	Path y nombre del .dbf de puntos. No se usa.
ShpFiles	Actualizacion de archivos shape	PointShpFile	C:\Inetpub\wwwroot\shp\Wupoints.shp	Path y nombre del archivo .shp de puntos
ShpFiles	Actualizacion de archivos shape	SpaceReplace	-	Carácter por el cual se reemplaza el carácter de espacio en blanco, para que el string que se pasa como parámetro a los programas de actualización de puntos y líneas, no se corte

Apéndice C - Resultados de los experimentos de planificación de rutas.

Se presentan a continuación los detalles de los experimentos realizadas con los algoritmos. El parámetro loadboundingbox que determina al zona de la ciudad a ser simulada tiene valor: 464731.500000 6137064.850078, 466698.718750 6139034.149922.

Los orígenes y destinos considerados se muestran en la Tabla C-1.

Número de prueba	Intersección de origen		Intersección de destino	
	1	YARO	ISLA DE FLORES	GALICIA
2	COLONIA	JUAN PAULLIER	DIECIOCHO DE JULIO	VAZQUEZ
3	CERRO LARGO	PIEDRA ALTA	JOAQUIN REQUENA	MALDONADO
4	MAGALLANES	DIECIOCHO DE JULIO	CANELONES	JOAQUIN REQUENA
5	TRISTAN NARVAJA	PAYSANDU	MINAS	SAN SALVADOR
6	CHANA	JUAN PAULLIER	LA PAZ	MINAS
7	DIECIOCHO DE JULIO	JUAN PAULLIER	DIECIOCHO DE JULIO	VAZQUEZ
8	JUAN JACKSON	DURAZNO	COLONIA	MARTIN MARTINEZ
9	JOSE ENRIQUE RODO	JOAQUIN REQUENA	URUGUAY	TACUAREMBO
10	SAN SALVADOR	YARO	URUGUAY	BARRIOS AMORIN
11	PALMAR	JUAN PAULLIER	CONSTITUYENTE	MINAS
12	EDUARDO VICTOR HAEDO	JOAQUIN REQUENA	TRISTAN NARVAJA	MERCEDES
13	SAN SALVADOR	SALTO	PAYSANDU	TACUAREMBO
14	EDUARDO VICTOR HAEDO	MARTIN MARTINEZ	CARLOS ROXLO	CONSTITUYENTE
15	GONZALO RAMIREZ	JUAN JACKSON	PAYSANDU	GERMAN BARBATO

Tabla C-1. Orígenes y destinos de las pruebas.

En el experimento con poco tráfico los parámetros considerados fueron los siguientes:

- Tiempo total simulado 10 horas.
- Ejecución por densidad 0.15 ± 0.05 .
- Estado estable 1 hora, tolerancia en la variación de vehículos 0. El estado estable se alcanzo a la 1:48:01 horas.

En la Tabla C-2 se muestran los resultados obtenidos.

Número de prueba	Planificación geográfica		Planificación por tiempo de viaje	
	Distancia (mts)	Tiempo (seg)	Distancia (mts)	Tiempo (seg)
1	1616	89,52	1633	89,07
2	1574	87,29	1580	86,28
3	1973	110,05	1975	110,05
4	1272	70,55	1280	70,34
5	1583	89,57	1586	88,96
6	1908	106,74	1908	106,74
7	1470	81,43	1470	81,43
8	1357	76,5	1367	76,26
9	1650	90,94	1671	90,9
10	1601	89,9	1618	89,45
11	1401	78,7	1401	78,7
12	766	42,32	766	42,32
13	1269	70,73	1269	70,73
14	1449	79,96	1464	79,7
15	1994	110,67	1996	110,67
Suma total	22883	1274.87	26976	1271.6

Tabla C-2. Resultados de las pruebas con poco tráfico.

En el experimento con mucho tráfico los parámetros considerados fueron los siguientes:

- Tiempo total simulado 10 horas.
- Ejecución por densidad 0.9 ± 0.05 .
- Estado estable 1 hora, tolerancia en la variación de vehículos 10%. El estado estable se alcanzo a la 1:24:01 horas.

En la Tabla C-3 se muestran los resultados obtenidos.

Número de prueba	Planificación geográfica		Planificación por tiempo de viaje	
	Distancia (mts)	Tiempo (seg)	Distancia (mts)	Tiempo (seg)
1	1616	521	1961	322,41
2	1574	515,56	1754	351,94
3	1973	611,08	2362	459
4	1272	522,95	1527	274,36
5	1583	596,1	1779	398,82
6	1908	855,61	2275	402,39
7	1470	420,26	1571	363,22
8	1357	547,11	1787	354,31
9	1650	481	1708	379,32
10	1601	600,64	1950	364,73
11	1401	489,09	1602	331,42
12	766	264	1269	227,57
13	1269	350,5	1361	348,3
14	1449	453,49	1772	339,13
15	1994	658,88	2246	387,66
Suma total	22883	7887.27	31416	5304.58

Tabla C-3. Resultados de las pruebas con mucho tráfico.