

CAPITULO 5
DESARROLLO E IMPLEMENTACION

5.1 Introducción

Una parte muy importante dentro de nuestro trabajo es la edición de las capas de la base de datos con las que vamos a trabajar. Nuestro objetivo es dar un tratamiento a los archivos shapefile para posteriormente exportarlos a la base de datos y trabajar con las capas que se encuentren almacenadas en esta base de datos.

En este capítulo se mostrará la manera en la que se editan los archivos shapefile, para posteriormente exportarlos a la base de datos y de esa manera recuperar y almacenar los datos de la base del volcán. Esto se realizará mediante el uso de herramientas comerciales desarrolladas por ESRI: ArcView, ArcInfo y ArcSDE.

Por otra parte, se mostrará el trabajo de software desarrollado, el cual es una aplicación web mediante la cual se pueden visualizar las capas del volcán en formato SVG. Esto se desarrolló con ayuda de trabajos anteriores, como el de Gerardo Cepeda [Cepeda-2003] que genera un archivo GML 2.1.2 con los datos recuperados de la base del volcán.

En la Figura 5 se muestra el diagrama de la arquitectura de nuestro trabajo, con los módulos de tratamiento y edición con herramientas especializadas y el módulo de visualización de capas en SVG. Además de tener como vínculo de estos dos módulos, el trabajo de Gerardo Cepeda, al generar los documentos GML versión 2.1.2 [Cepeda-2003].

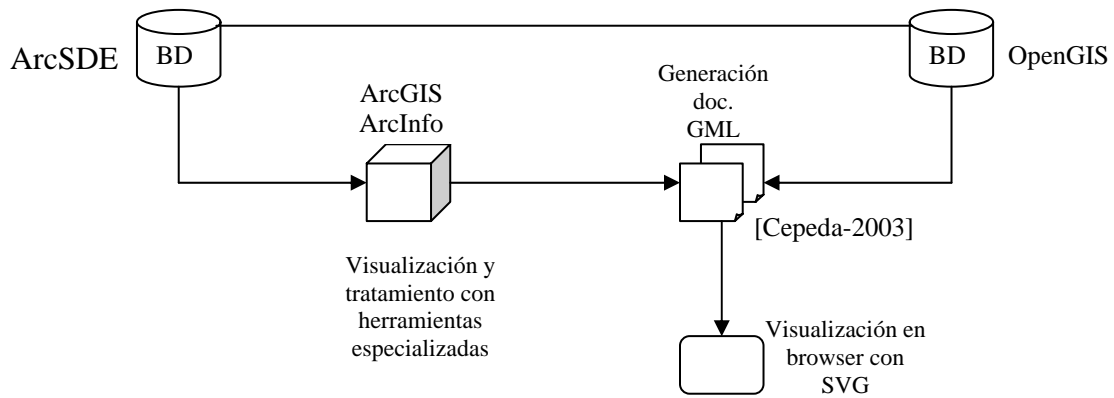


Figura 5. Diagrama de la arquitectura.

5.2 Requerimientos

Los requerimientos del sistema fueron determinados en base a reuniones que se tuvieron entre diferentes tesis y profesores del departamento de sistemas, entre los cuales podemos mencionar al doctor David Sol Martínez, al maestro Antonio Felipe Razo Rodríguez, al tesista Gerardo Cepeda Porras, a los desarrolladores de este trabajo, Benjamín Rodríguez Ortega y Gustavo Alfredo Negrete López, entre otros. La idea principal surgió de un proyecto que se tenía en el departamento de sistemas que consistía en desarrollar herramientas para el acceso y la visualización de información cartográfica, dado que el proyecto es considerablemente grande se decidió dividirlo en varios proyectos de tesis, entre los cuales se encuentra la presente tesis.

A continuación se muestra en la Figura 5.1 un diagrama general del flujo de información de la parte del proyecto que se relaciona directamente con esta tesis.

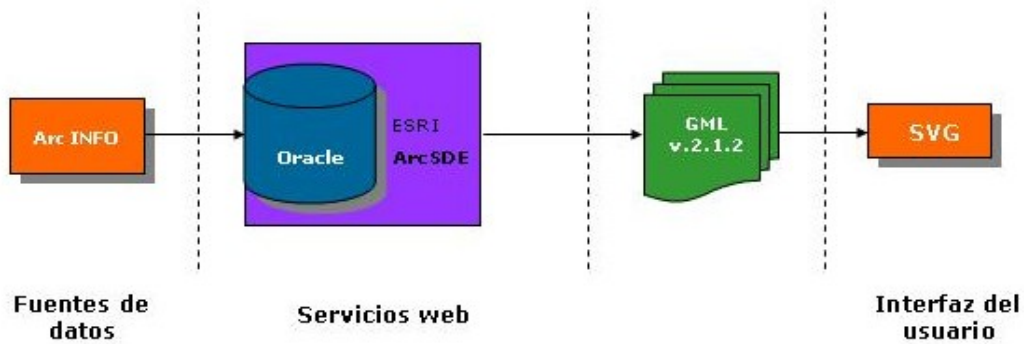


Figura 5.1. Flujo de datos de la arquitectura.

En el diagrama primeramente tenemos la fuente de datos que es la herramienta comercial ArcInfo, la cual nos provee de los datos iniciales, posteriormente se tiene el almacenamiento de estos datos en una base de datos ArcSDE, para después ser transformados a un formato GML 2.1.2 con la ayuda de la tesis de Gerardo Cepeda [Cepeda-2003], la cual nos produce los datos de entrada de nuestro proyecto. Una vez teniendo estos datos en formato GML 2.1.2 podemos pasar a la parte que consiste en transformar esos datos a un formato SVG, que es a grandes rasgos lo que realiza nuestra aplicación.

Basados en las opiniones y comentarios de los integrantes de este proyecto se logró delimitar los requerimientos que a continuación se mencionan:

Requerimientos de entorno, requerimientos, tanto de hardware como de software, necesarios para el funcionamiento del proyecto.

- API de Java 1.4.2, para la conversión de GML a SVG.
- Internet Explorer 6 o superior, para la ejecución del software.
- Computadora PC, para la utilización del software, dado que se utiliza Internet Explorer 6.
- Tomcat 4.1.1.8, para desplegar los servicios del proyecto.
- Tesis de Gerardo Cepeda, para la generación de documentos GML [Cepeda-2003].
- Librerías de servlets, taglibraries, para el desarrollo de la aplicación.
- Herramientas comerciales, para el tratamiento de la cartografía.

Requerimientos ergonómicos, requerimientos de la interfaz humano-computadora.

- Soportar la internacionalización de mensajes para que el sitio también pueda ser accesible a personas que no dominen el idioma español.
- Tener facilidad de uso.
- Sitio amigable al usuario.
- Presentar información espacial en formato gráfico.

Requerimientos funcionales, nos describen lo que el sistema deberá hacer:

- Realizar la visualización de documentos GML, permitiendo elegir un orden de visualización en las capas o documentos a visualizar.
- Realizar la conversión de documentos GML a SVG.
- Permitir al usuario subir documentos GML (previamente validado por los esquemas de Gerardo Cepeda) para ser visualizados.
- Implementación del sistema en un patrón de diseño de Modelo-Vista-Controlador.

Restricciones de diseño, requerimientos relacionados con estándares que el sistema debe cumplir:

- Uso de estándar GML 2.1.2 de OpenGIS (basado en los esquemas de Gerardo Cepeda).
- Uso del formato abierto de SVG para la visualización gráfica.
- Independencia de plataforma.
- Servicio accesible vía Internet.

5.3 Arquitectura

Como ya se había mencionado anteriormente la arquitectura está basada en el Modelo-Vista-Controlador (MVC), para lo cual se utilizarán las librerías que fueron desarrolladas por Jakarta, Struts 1.1. Estas librerías nos permitirán la implementación del modelo MVC de una manera más transparente, al mismo tiempo que nos permitirán

agregar elementos de internacionalización para los lenguajes de inglés y español [Struts-2003].

Una vez que la información ha pasado por los módulos de edición y almacenamiento de las herramientas comerciales de ESRI, y por el módulo de generación de GML a través del trabajo de Gerardo Cepeda [Cepeda-2003], el flujo de información del sistema para la visualización de SVG es el siguiente:

- 1.- El usuario envía la solicitud de que documentos deben graficarse y en que orden o envía la solicitud de subir un documento GML al sistema, en caso de que deba graficarse se utilizan las clases del paquete svgGen. Por otro lado si debe subirse un documento GML, se utiliza la clase visualizaAction programada para la aplicación web.
- 2.- El controlador identifica la solicitud y realiza un mapeo para delegar la responsabilidad de esa tarea al bean correspondiente. En este paso solamente se utilizan las clases de Struts y el archivo de configuración “struts-config.xml”.
- 3.- Una vez que el bean recibe la acción, éste valida la información y realiza la acción en caso de que la información sea válida, de lo contrario indica el error de validación. Aquí se utilizan las clases de visualizaAction y showAv_gmlForm para la validación y para la visualización.

4.- Ya realizada la acción por el bean, el controlador decide la manera de presentar dicho resultado al usuario a través de los componentes de visualización (JSP's). Aquí solamente se utilizan las clases de Struts para direccionar al jsp correspondiente.

A continuación se muestra en la Figura 5.2 una ilustración más general de cómo funciona el MVC con ayuda de Struts 1.1.

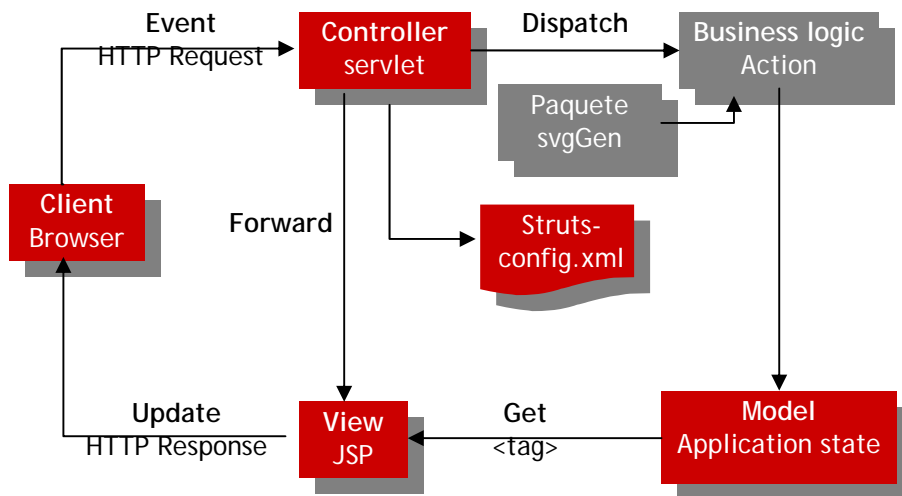


Figura 5.2. Arquitectura Struts.

En el diagrama anterior tenemos la caja de “Client” (que es donde empieza el diagrama), esta parte del diagrama está identificada por un browser o explorador de Internet a través del cual el usuario deberá interactuar para poder acceder y usar el software. Cabe mencionar que para este trabajo se usó Internet Explorer en su versión 6 dado que los

scripts fueron específicamente diseñados para este explorador, bajo una plataforma PC. En seguida se realiza una petición (“Request”) a través del explorador. Esto nos lleva a pasar el control de la aplicación al controlador del servlet que a continuación se describe.

El controlador del servlet (la caja de “Controller servlet”) es un servlet que es el encargado de manejar las peticiones y tomar las decisiones de que acciones deben tomarse para resolver la petición del usuario. Las acciones se toman en base a las peticiones que están en un archivo de configuración llamado “struts-config.xml”. Una vez identificada la acción que se debe tomar se manda a llamar al bean correspondiente para ejecutar dicha acción, esto es en la caja de “Business Logic”. Es aquí donde entra nuestro paquete de generación de documentos SVG. Una vez que se ejecuta la acción esta debe modificar el estado de la aplicación para posibles peticiones posteriores, la modificación del estado de la aplicación se hace en la caja de “Model”, que a su vez cede el control de la aplicación a una caja de visualización “View”, la cual decide como presentar al usuario el resultado de la petición a través del explorador, volviendo nuevamente al componente inicial que fue la caja de “Client” representada por el explorador.

A continuación describiremos el paquete de generación de documentos SVG y sus clases. Este paquete fue creado para transformar documentos GML, basados en los esquemas de Gerardo Cepeda [Cepeda-2003], a un formato SVG. Aquí se describirán las funciones de las clases utilizadas. En el Apéndice G podemos encontrar los diagramas UML estas clases:

- **bBox**: Esta clase permite almacenar las coordenadas del bounding box originales, es decir, sin escalar con las proporciones establecidas, así como también nos permite obtener la proporción con la que las coordenadas del documento serán transformadas. Esta clase contiene los siguientes métodos: `getProportion` y `transformBBoxCoor`.
- **bBoxReader**: Clase que nos permite leer el bounding box de un documento específico. Esta clase contiene los siguientes métodos: `getBBox` y `setNamespace`.
- **descDataReader**: Clase que nos permite leer la información descriptiva (no geométrica) de los documentos GML. Esta clase contiene el siguiente método: `getDesData`.
- **docColor**: Clase que no permite definir los colores con los cuales serán dibujadas las geometrías, si no se especifica un color por default se tienen los siguientes colores: Puntos-> Interior: Azul: "#040ED3", Exterior: Azul claro: "#6066D2". Líneas-> Interior: "none", Exterior: Azul claro: "#538de0". Polígonos-> Interior: Gris claro: "#e5e5e5", Exterior: Gris oscuro: "#9e9e9e". Esta clase contiene los siguientes métodos: `setLineColors`, `setPointColors`, `setPolygonColors`.
- **docReader**: Clase que nos permite leer un documento XML del disco duro. Esta clase contiene el siguiente método: `readDoc`.

- `JpgCreator`: Clase que nos permite generar archivo de imagen tipo `jpg`, de los archivos generados tipo `SVG`. Esta clase sólo soporta archivos `SVG` de tamaño muy limitado por lo que no se usa en este proyecto. Esta clase contiene el siguiente método: `createJpg`.
- `lineReader`: Clase que nos permite la lectura de las líneas en un documento tipo `GML`, basado en los esquemas de Gerardo Cepeda. Esta clase contiene el siguiente método: `readLines`.
- `mLineReader`: Clase que nos permite la lectura de las multilíneas en un documento tipo `GML`, basado en los esquemas de Gerardo Cepeda. Esta clase contiene el siguiente método: `readMLines`.
- `mPolygonReader`: Clase que nos permite la lectura de multipolígonos en un documento tipo `GML`, basado en los esquemas de Gerardo Cepeda. Esta clase contiene el siguiente método: `readMPolygons`.
- `pointReader`: Clase que nos permite la lectura de puntos en un documento tipo `GML`, basado en los esquemas de Gerardo Cepeda. Esta clase contiene el siguiente método: `readPoints`.

- `polygonReader`: Clase que nos permite la lectura de polígonos en un documento tipo GML, basado en los esquemas de Gerardo Cepeda. Esta clase contiene el siguiente método: `readPolygons`.
- `proportion`: Clase que nos permite establecer la proporción con que las coordenadas del GML serán transformadas.
- `resolution`: Clase que no permite establecer la resolución del SVG a generar a partir del GML. Esta clase contiene los siguientes métodos: `setX`, `setY`, `getX`, `getY`.
- `settings`: Clase que nos permite almacenar las características con que se leerá el documento GML, basado en los esquemas de Gerardo Cepeda. Esta clase contiene los siguientes métodos: `getLines`, `getMLines`, `getMPolygons`, `getPoints`, `getPolygons`, `readGeom`, `setName`, `setRenderLines`, `setRenderMPolygons`, `setRenderPoints`, `setRenderPolygons`, `setSize`.
- `svgGen`: Clase que nos permite mandar a llamar todos los métodos necesarios para la generación de documentos SVG. Esta clase no contiene métodos, ya que con los constructores se realiza todo el proceso.
- `svgMaker`: Esta clase se encarga de generar el archivo SVG, que nos permitirá visualizar gráficamente los documentos GML. Esta clase contiene el siguiente método: `createSVG`.

Para mas información en el apéndice G se pueden encontrar los diagramas UML de las clases que fueron creadas como parte de la capa lógica para la conversión de GML a SVG.

5.4 Servicio de ArcSDE

En nuestro trabajo, es necesario tener el servicio de ArcSDE activo. Para levantar el servicio de ArcSDE, primero hay que conectarse a la máquina donde se encuentra instalado. Esto se logra desde cualquier máquina que esté conectada a la red. Se ejecuta el comando “sdemon -o start” con el que se inicia el servicio de ArcSDE. Al ejecutar este comando, se va a requerir escribir la contraseña para iniciar el servicio.

Para saber que el servicio de ArcSDE está activo, debemos teclear el comando “sdemon -o status”, de esta manera sabemos que el servicio está funcionando y que podemos trabajar con él. Todo lo relacionado con el almacenamiento y recuperación de datos espaciales mediante ArcSDE, lo podemos ver a detalle en el trabajo de Rafael Ruiz [Ruiz-2002].

5.5 Edición de mapas

ArcMap, al igual que ArcView, son visualizadores de ESRI, con los cuales podemos desplegar los datos geográficos. A su vez, estas herramientas nos dan el poder de editar las capas de diferentes modos, ya sea creando nuevos features, editando los features ya existentes, editando sus atributos, o realizando consultas sobre la capa. Todo esto con la ayuda de diversas funciones de edición.

Para poder editar una capa, lo primero que se tiene que hacer es iniciar la aplicación de ArcMap, que se encuentra donde se instaló ArcInfo. La ventana que se muestra en la Figura 5.3, es la interfaz de ArcMap, la cual contiene varias barras de herramientas. Del lado izquierdo se mostrarán los nombres de las distintas capas que se desplegarán en el lado derecho de la pantalla.

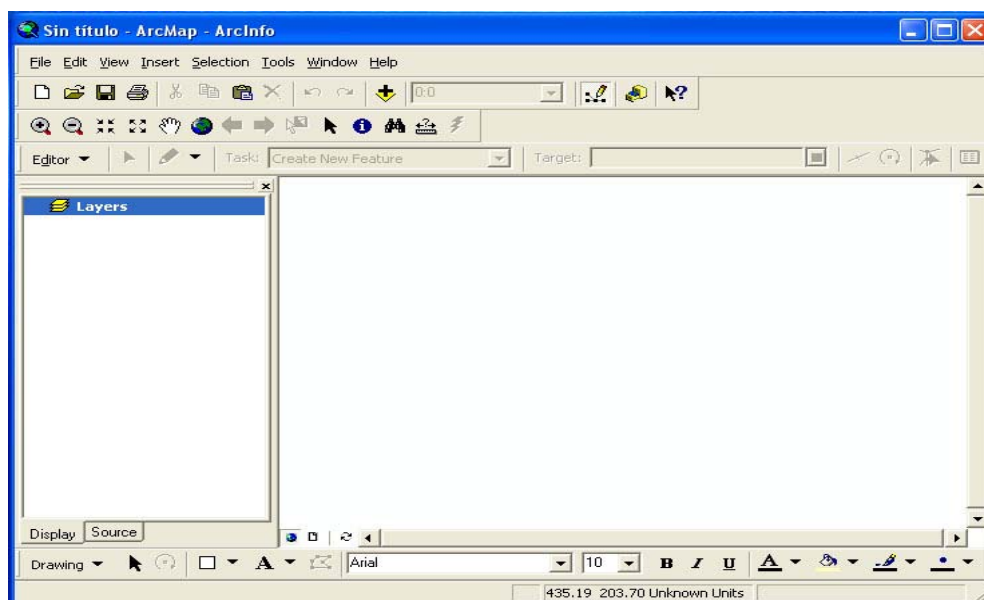


Figura 5.3. Interfaz de ArcMap.

A continuación se muestra de manera general el modo en que usamos ArcMap para editar las capas con las cuales vamos a trabajar. Una vez iniciada la aplicación de ArcMap, debemos añadir la capa que deseamos editar. También se puede agregar la capa desde ArcCatalog, con tan sólo arrastrarla a ArcMap.

Toda tarea de edición tiene lugar en una sesión de edición. Para comenzar, debemos escoger Star Editing del menú de Editor. Los cambios que hagamos son visibles inmediatamente en el mapa, pero no son salvados sino hasta que nosotros decidamos, o hasta que queramos terminar la sesión de edición con Stop Editing.

ArcMap cuenta con una gran cantidad de comandos de edición, con los que podemos realizar todo tipo de operaciones sobre los mapas, podemos fusionar features de una misma capa a un solo feature, con el comando Merge; podemos combinar features de diferentes capas a una sola capa, con el comando Union; y podemos crear nuevos features a partir de features que tienen áreas en común, gracias al comando Intersect. También podemos completar features o hacer correcciones al feature. En la Figura 5.4 mostramos un ejemplo del comando Union en dos de las capas que se editaron. El resultado de la unión son 4 features, los cuales son el producto de combinar los features que se encontraban dentro de ellos inicialmente.

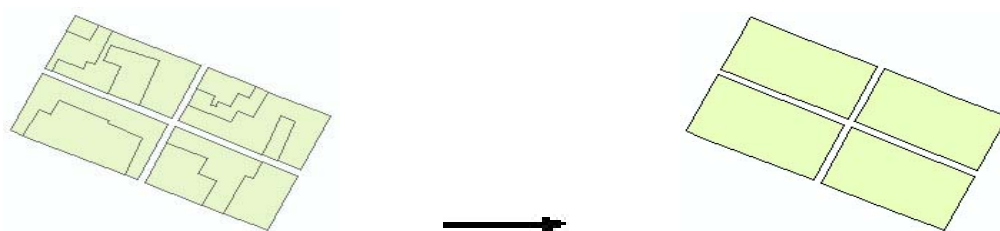


Figura 5.4. Unión de features.

5.5.1 Trabajando con tablas y atributos

Una tabla es un componente de la base de datos que representa features geográficos y nos muestra los atributos de los mismos. Para mostrarla, en la tabla de contents, hacemos clic con el botón derecho del Mouse sobre la capa de la cual queremos ver su tabla. Hacemos clic en Open Attribute Table y la tabla de la capa es desplegada, como se muestra en la Figura 5.5.

FID	Shape*	AREA	PERIMETER	IDS	OBJETO	NOMBRE
0	Polygon	1430.40041	161.40427	3209	CONSTRUCCIO	
1	Polygon	2675.54202	252.12321	3209	CONSTRUCCIO	
2	Polygon	688.71668	109.26499	3209	CONSTRUCCIO	
3	Polygon	2210.10813	218.35398	3209	CONSTRUCCIO	
4	Polygon	1203.21713	144.12537	3209	CONSTRUCCIO	
5	Polygon	1114.74222	134.88259	3209	CONSTRUCCIO	
6	Polygon	822.74143	140.69804	3209	CONSTRUCCIO	
7	Polygon	583.99308	107.3544	3209	CONSTRUCCIO	
8	Polygon	407.23992	80.99322	3209	CONSTRUCCIO	
9	Polygon	1550.27899	159.16244	3209	CONSTRUCCIO	
10	Polygon	607.84347	99.45593	3209	CONSTRUCCIO	
11	Polygon	9470.57297	600.24658	3209	CONSTRUCCIO	
12	Polygon	621.31065	105.82524	3209	CONSTRUCCIO	
13	Polygon	3032.89567	324.38398	3209	CONSTRUCCIO	
14	Polygon	2917.96942	218.43176	3209	CONSTRUCCIO	
15	Polygon	910.44668	132.32612	3209	CONSTRUCCIO	
16	Polygon	1182.96626	177.89987	3209	CONSTRUCCIO	
17	Polygon	1259.40052	157.44761	3209	CONSTRUCCIO	
18	Polygon	9717.65087	695.04935	3209	CONSTRUCCIO	

Figura 5.5. Tabla de atributos de la capa pobau_mod.

Nuestra base de datos es solamente tan buena como la información que contiene. En nuestro trabajo pudimos actualizar los atributos de los features desplegados en los mapas. Con ArcMap podemos editar cualquier valor de los atributos que aparecen en la tabla, así como añadir nuevos campos a las tablas o eliminar algunos ya existentes. Al añadir un campo, debemos especificar el nombre del mismo, y el tipo de dato que va a manejar. Podemos eliminar un campo de una tabla, al hacer clic con el botón derecho del Mouse en el encabezado del campo, y escoger Delete Field.

Los atributos de una capa pueden ser creados o editados en la ventana de diálogo de Attributes. Después de seleccionar el feature sobre el cual queremos editar sus atributos, hacemos clic en el botón de Attributes en la barra de herramientas de Editor y nos despliega la ventana de diálogo que se muestra en la Figura 5.6.

Para añadir o modificar un valor de un atributo de un feature, debemos hacer clic en el campo del feature que se encuentra del lado izquierdo de la ventana y hacer los cambios en la columna de Value del lado derecho.

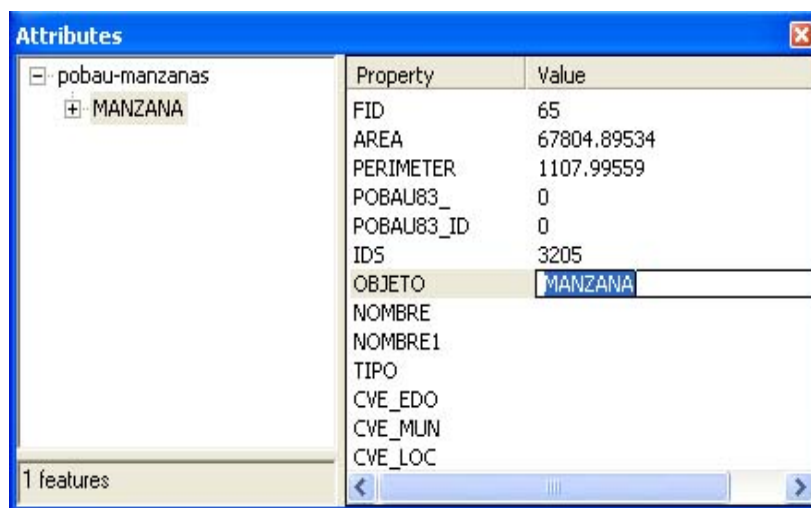


Figura 5.6. Ventana de atributos.

5.5.2 Selección de features para generar capas nuevas

Seleccionando los features sobre la capa, identificamos el feature que deseamos editar. Por ejemplo, antes de mover, eliminar o copiar un feature, lo debemos seleccionar. Además de que también lo debemos seleccionar antes de poder ver sus atributos.

En la Figura 5.7 se muestra un ejemplo de selecciones de features.

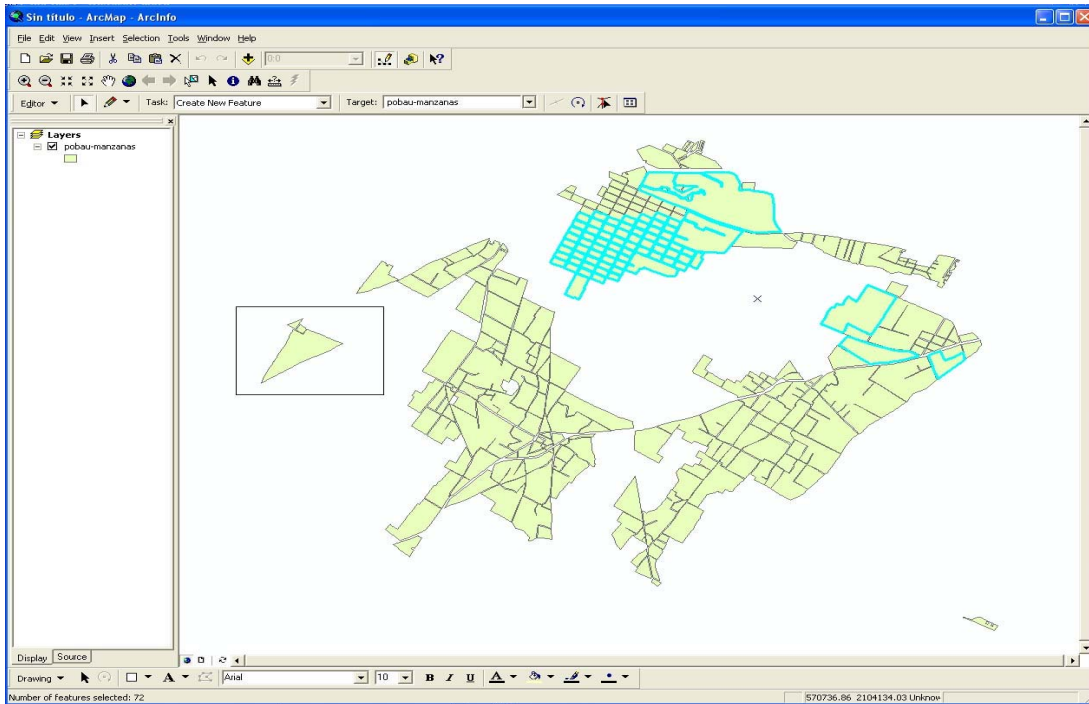


Figura 5.7. Selección de features en un mapa.

Podemos seleccionar features de diversas maneras, una de ellas es con Editor, haciendo clic sobre ellos, otra manera, es creando una línea o un polígono que intersecte los features que queremos editar. Si necesitamos seleccionar más de un feature, dejamos presionado Shift mientras hacemos clic sobre los features. Otra manera es por medio de la tabla de atributos de la capa, seleccionando en ella el feature.

También se pueden hacer selecciones de features, tomando en cuenta sus atributos o su ubicación. Una vez hecha la selección, ésta puede ser exportada como una capa por separado, mostrando únicamente los features seleccionados.

Structured Query Language (SQL) es un poderoso lenguaje en el contexto de bases de datos que usamos para definir diversos criterios que pueden consistir en atributos, operadores y cálculos. Cuando realizamos una búsqueda con expresiones en SQL, podemos seleccionar features o records de las tablas. Este tipo de expresiones las utilizamos al seleccionar features por sus atributos. Esto lo logramos haciendo clic en el campo de Selection en Select By Attributes. Inmediatamente aparecerá una ventana en la que se pueden desarrollar los queries, escogiendo la capa y el tipo de selección que deseamos. En la ventana nos aparecerán los campos de la capa elegida, y los diferentes valores de los campos, dependiendo el campo que se escoja. También nos aparecerá una serie de operadores con los cuales podemos desarrollar diferentes expresiones en SQL. El cuadro donde se forma la expresión en versión de SQL, nos da la libertad de poder escribir nosotros mismos los queries que queremos aplicar a la capa. La sintaxis del query va a depender del origen del tipo de dato que se está consultando.

La Figura 5.8 muestra la ventana en la que desarrollamos un query sobre la capa pobau_mod, en la que seleccionamos el campo Objeto, y queremos obtener todos los features que sean Manzanas, sin tomar en cuenta a los features que son Construcciones.

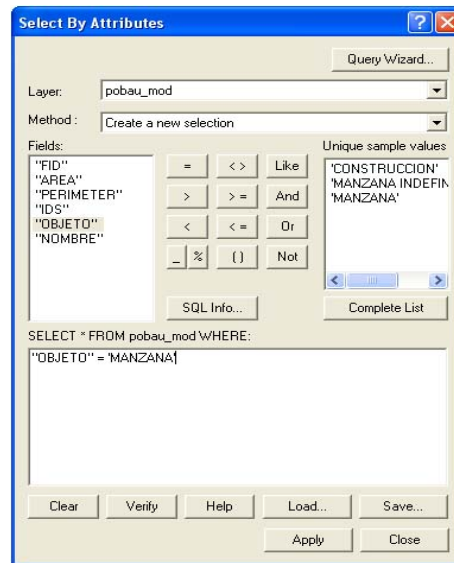


Figura 5.8. Query sobre la capa pobau_mod.

Otra manera de realizar consultas, es con queries espaciales. En este tipo de consultas queremos encontrar features basados en lo referente a su ubicación espacial. Al combinar queries, podemos realizar consultas más complejas. Se pueden encontrar nuevas relaciones espaciales cuando se empiezan a hacer preguntas como: Dónde...?, Dónde está el más cerca?,Cuál es la intersección?. Este tipo de consulta lo realizamos haciendo clic en el campo de Selection en Select By Location. Se desplegará una ventana en la que debemos especificar la operación que queremos realizar:

1. La capa sobre la cual vamos a seleccionar los features.
2. El método de selección, como puede ser que se intersecte, que tenga su centro en, que lo contenga, que comparta un segmento, entre otros.
3. La capa que se utiliza para la búsqueda de los features.

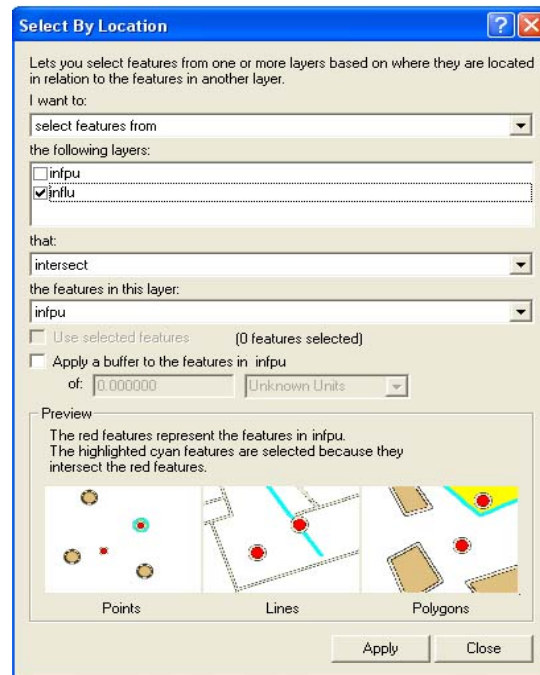


Figura 5.9. Query espacial de intersección de capas.

En la Figura 5.9 se muestra una consulta espacial sobre los features de la capa influ, la cual contiene geometrías de polilíneas que representan puentes y líneas eléctricas, y los features de la capa infpu, la cual contiene puntos que representan torres eléctricas, escuelas y templos. El resultado de la consulta selecciona los features de la capa influ que intersectan a los de la capa infpu

Una vez hecha la selección de los features que cumplan con los criterios deseados, podemos trabajar con estos features en una capa independiente. Podemos exportar los features seleccionados a un nuevo archivo shapefile o a un feature class de una base de

datos. La forma de hacer la exportación es haciendo clic con el botón derecho del Mouse sobre la capa seleccionada, en Data y en Export Data. En la Figura 5.10 se muestra la exportación de los features seleccionados de puentes y líneas eléctricas de la capa influ.

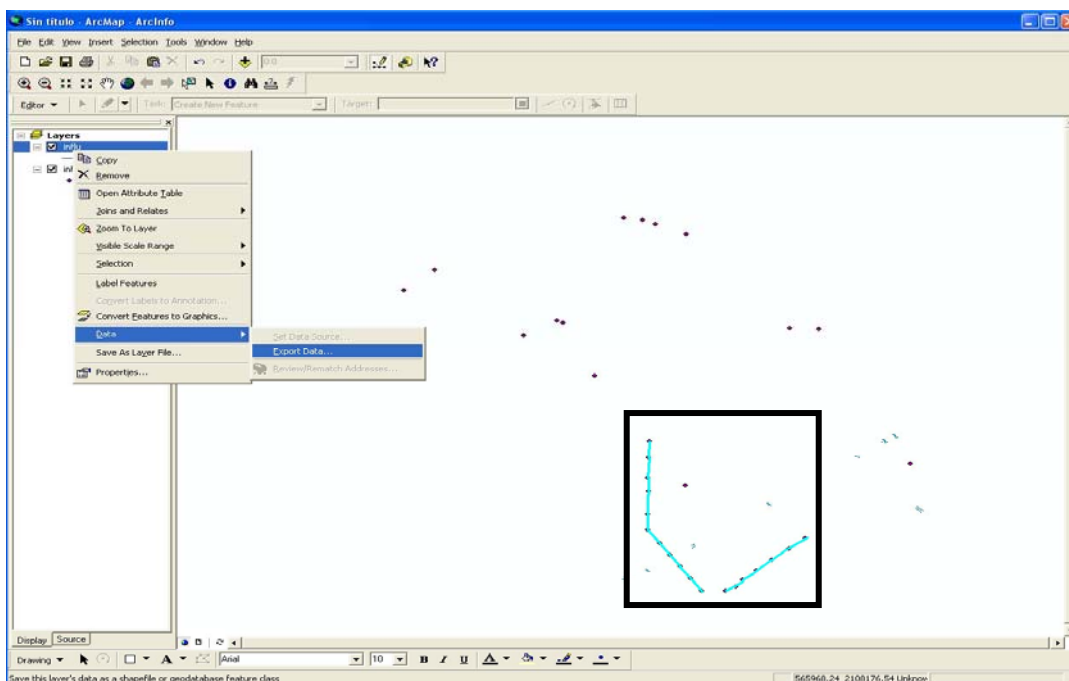


Figura 5.10. Exportar selección.

En ese momento, parecerá una ventana de diálogo como la de la Figura 5.11, en la cual especificamos que queremos exportar los features seleccionados, y damos la ruta y el nombre del archivo de salida, ya sea un shapefile o un feature class.

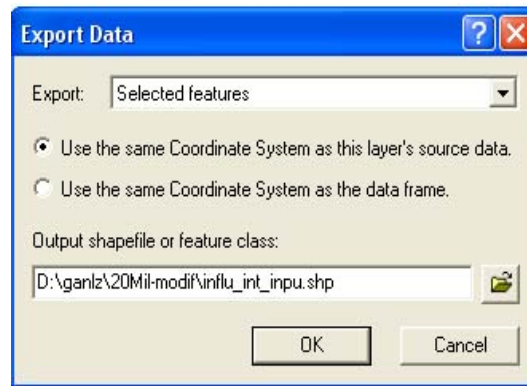


Figura 5.11. Ventana de Export data.

Una vez hecha la edición y actualización de las capas con las que vamos a trabajar, debemos exportarlas a la base de datos del volcán para posteriormente hacer su visualización en SVG.

5.6 Almacenamiento de datos

Mediante el uso de ArcSDE, otras herramientas desarrolladas por ESRI, como es el caso de ArcCatalog pueden tener acceso a la base de datos. Este módulo es parte de ArcInfo, el cual es el encargado de manejar los datos, y las conexiones con la base de datos mediante ArcSDE.

Mientras que ArcSDE es el encargado de la administración de los datos, ArcCatalog es el encargado de descargar los datos a la base mediante el uso de ArcSDE. En caso de que

no exista una conexión a una base de datos, ArcCatalog se encarga de manejar los diferentes tipos de archivos que manejan los visualizadores de ESRI. Entre los visualizadores tenemos a ArcView y ArcMap, que es otro módulo de ArcInfo, en los cuales se pueden visualizar y editar los datos geográficos.

Para almacenar los datos en la base, el primer paso es iniciar la aplicación de ArcCatalog que se encuentra donde se instaló ArcInfo. La ventana mostrada en la Figura 5.12 será desplegada. Esta es la interfaz de ArcCatalog, la cual contiene varias opciones, por ejemplo, agregar una conexión a una base de datos. En este caso tenemos dos conexiones trabajando en el servidor “centia.udlap.mx”, y nos muestra que es una conexión a una base de datos espacial mediante ArcSDE. También nos muestra las fuentes de datos del lado izquierdo de la pantalla, y los datos que están guardados en ese destino, los muestra del lado derecho.

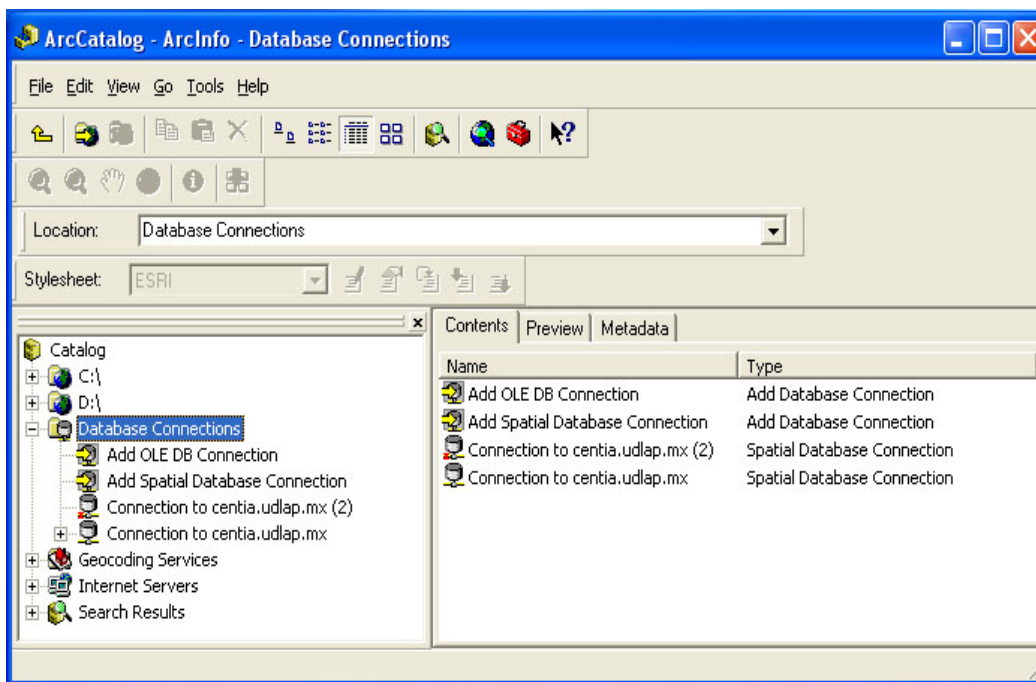


Figura 5.12. Interfaz de ArcCatalog.

Una vez iniciada la aplicación de ArcCatalog, se debe seleccionar Database Connections que nos muestra las diferentes conexiones que existen. Estas conexiones son al servicio de ArcSDE, que es el encargado de exportar los datos de un archivo shapefile, a la base.

Existen dos formas de llamar al comando de exportar shapefiles a la base de datos; una es mediante las herramientas ArcToolBox, y otra es con el botón derecho del Mouse. Las Figuras 5.13 y 5.14 muestran de manera clara como se selecciona este comando:

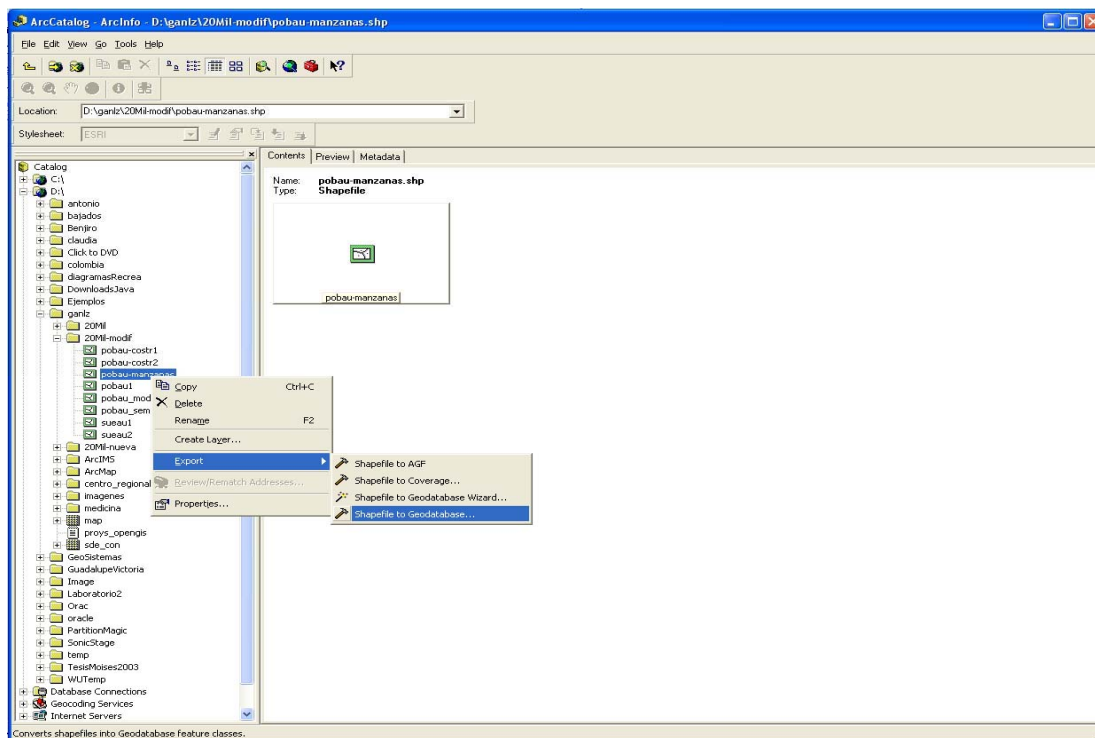


Figura 5.13. Exportar de Shapefile a Geodatabase con el Mouse.

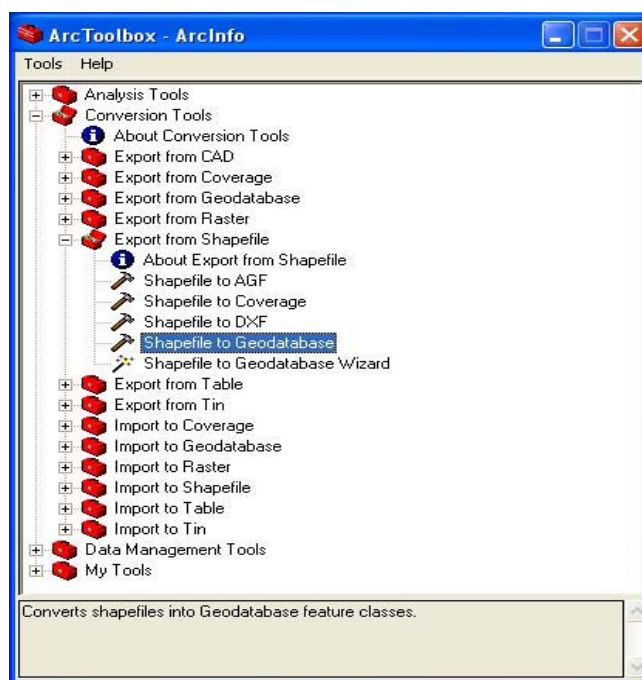


Figura 5.14. Exportar de Shapefile a Geodatabase con ArcToolBox.

Una vez seleccionada la opción de “Shapefile to Geodatabase”, se despegará una ventana de diálogo, la cual es la encargada de recibir los parámetros del archivo que será exportado a la base de datos. Se le asigna un nombre mediante el cual se hace la transformación al formato soportado por ArcSDE. Con sólo dar un refresh a la ventana donde se muestran los datos de cada destino, se da por terminado la exportación de los datos. Esto se tiene que hacer para cada uno de los shapefiles. La ventana de diálogo de “Shapefile to Geodatabase” se muestra en la Figura 5.15.

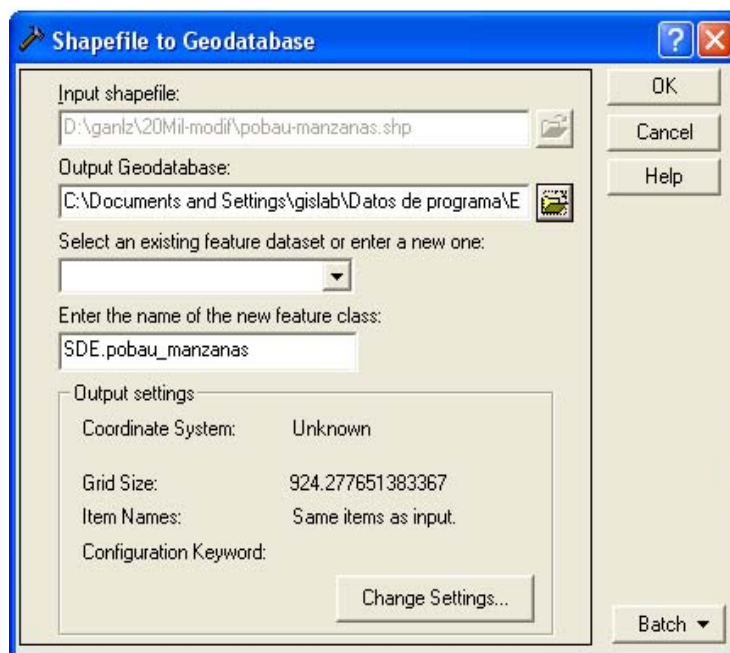


Figura 5.15. Ventana para exportar de Shapefile a Geodatabase con ArcSDE.

5.7 Recuperación de capas mediante ArcCatalog y ArcSDE

ArcCatalog es una herramienta que nos sirve para almacenar los datos en la base de datos del volcán, pero a su vez también sirve para recuperarlos mediante ArcSDE. A continuación se muestra como se realiza la recuperación de una capa mediante ArcCatalog, y lo mismo se tendrá que realizar para las demás capas.

En primer lugar se hace la conexión a la base de datos mediante ArcSDE. En nuestro caso se hace la conexión al servidor “centia.udlap.mx”, al servicio “5151”, a la base

de datos del volcán “sdedb”; a la cuenta del usuario “sde” y su password. Esta conexión a la base de datos del volcán se muestra en la Figura 5.16.

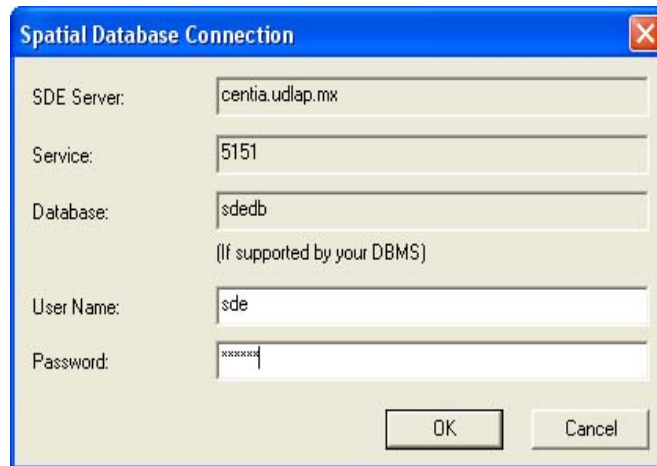


Figura 5.16. Conexión a la base de datos del volcán en ArcCatalog.

Una vez establecida la conexión con la base de datos, se despliegan todas las capas que se encuentran almacenadas en la base del volcán. La Figura 5.17 muestra las capas disponibles en la base de datos de ArcSDE. Cada una de ellas tiene un ícono que nos señala su tipo de dato, ya sea punto, línea, polígono, u otra clase de objeto. También nos indica que clase de tipo de objeto son, en este caso se puede observar que todos son features classes. Cada feature class corresponde a un tipo de archivo shapefile.

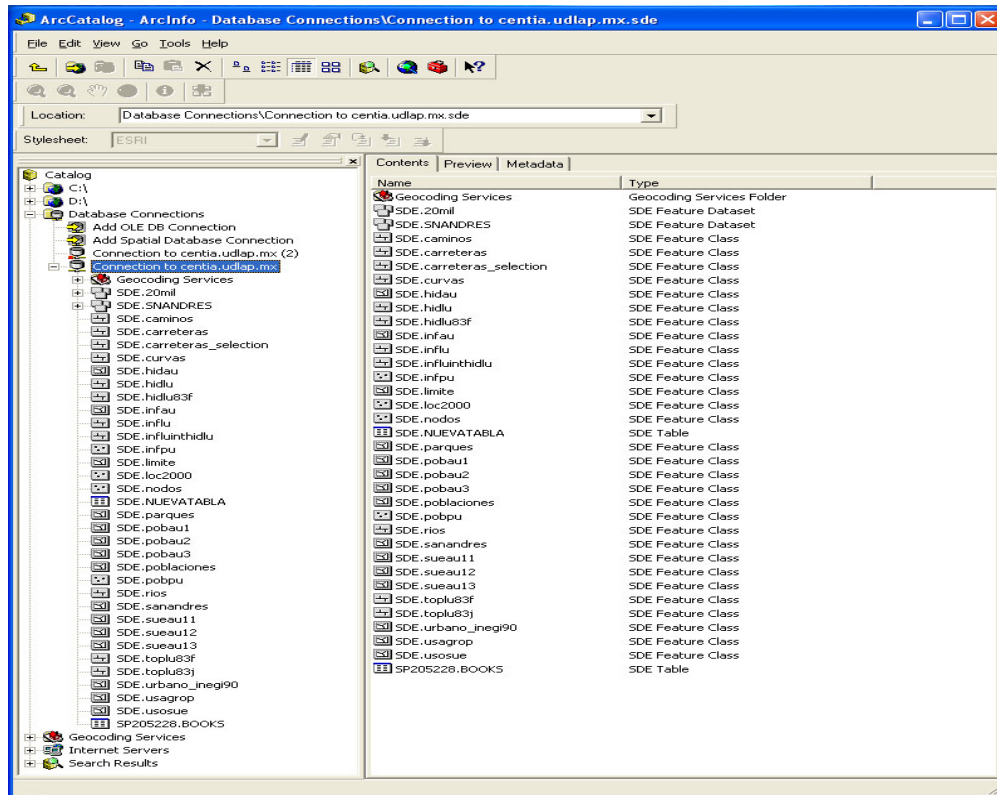


Figura 5.17. Capas de la base de datos del volcán en ArcCatalog.

Una vez que están todas las capas listadas, podemos seleccionar una para poder conocer sus propiedades, haciendo doble clic sobre ella. En la ventana que se despliega se pueden apreciar las propiedades de la capa, entre las que destacan, los campos descriptivos de la capa.

Con ArcCatalog se pueden visualizar las capas recuperadas, se pueden visualizar de modo geográfico o descriptivo, en forma de tabla; esto es dando clic en la caja de Preview y escogiendo la opción de Geography o Table. Sin embargo, no se pudo realizar ninguna

consulta o query sobre ella. Este tipo de operaciones como ya se mencionó antes, se realizan con otras aplicaciones como ArcView y ArcMap.

5.8 Acceso local mediante ArcSDE y generación de GML

Para poder utilizar el servicio de ArcSDE, fuera de las herramientas comerciales de ESRI, se han desarrollado trabajos previos, como el de Rafael Ruiz [Ruiz-2002], con el cual se conecta a ArcSDE. El usuario puede consultar la información que se encuentra en la base de datos del volcán, es decir, se desarrolló una forma de acceder a los datos sin tener que contar con las herramientas de ESRI. Para poder acceder a la base datos, y posteriormente generar los archivos GML de las capas, utilizamos el trabajo desarrollado por Gerardo Cepeda [Cepeda-2003].

Para poder acceder a la base de datos del volcán, se tiene que proporcionar, el servidor, el puerto, el usuario, el password y la base de datos, como se muestra en la Figura 5.18. En caso de que uno o varios de los parámetros sean incorrectos, aparece un mensaje de error. Una vez realizada la conexión mediante ArcSDE, se pueden consultar las diversas capas almacenadas en la base de datos.



Figura 5.18. Conexión a la base de datos.

La Figura 5.19 nos muestra las capas que se tienen disponibles en la base del volcán, las cuales fueron almacenadas en ArcCatalog y ArcSDE.

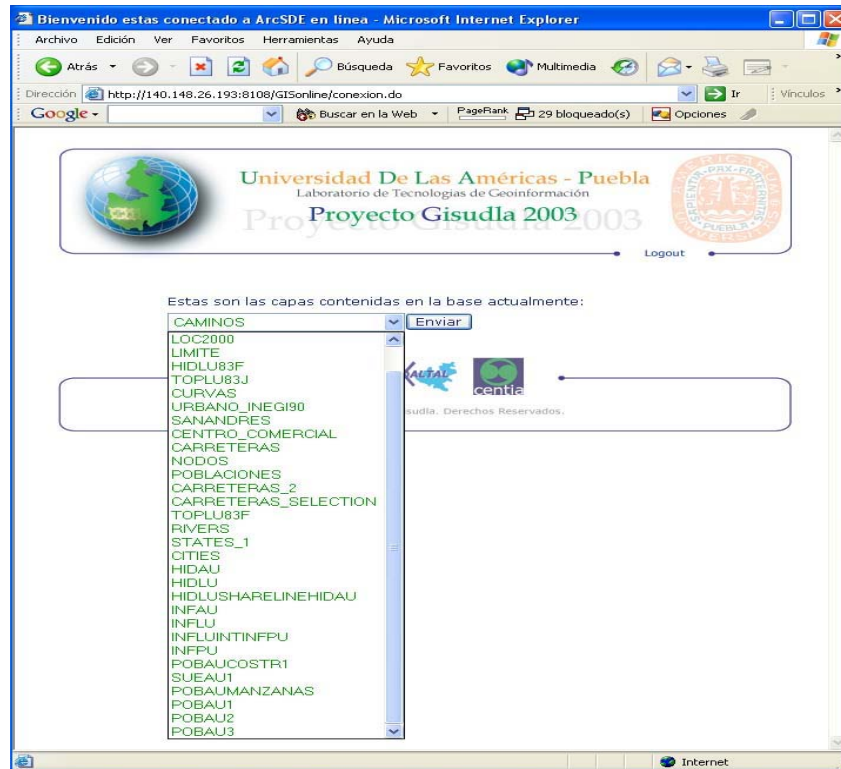


Figura 5.19. Capas de la base de datos del volcán.

Una vez seleccionada la capa con la que queremos trabajar, generamos el documento GML en su versión 2.1.2, y éste posteriormente lo descargaremos en un archivo ZIP, como lo muestra la Figura 5.20. Esto se traduce a un documento SVG con nuestra aplicación, y se genera la visualización de las capas de la base de datos del volcán.



Figura 5.20. Documento GML creado, opción de descargar.

5.9 Aplicación web de visualización en SVG

La primera página de nuestra aplicación web, se muestra en la Figura 5.21. Básicamente es la presentación de nuestro trabajo, donde hay ligas que mandan a los Documentos, Java Docs, UML, Software, donde se puede empezar a trabajar con la aplicación visualizando capas.

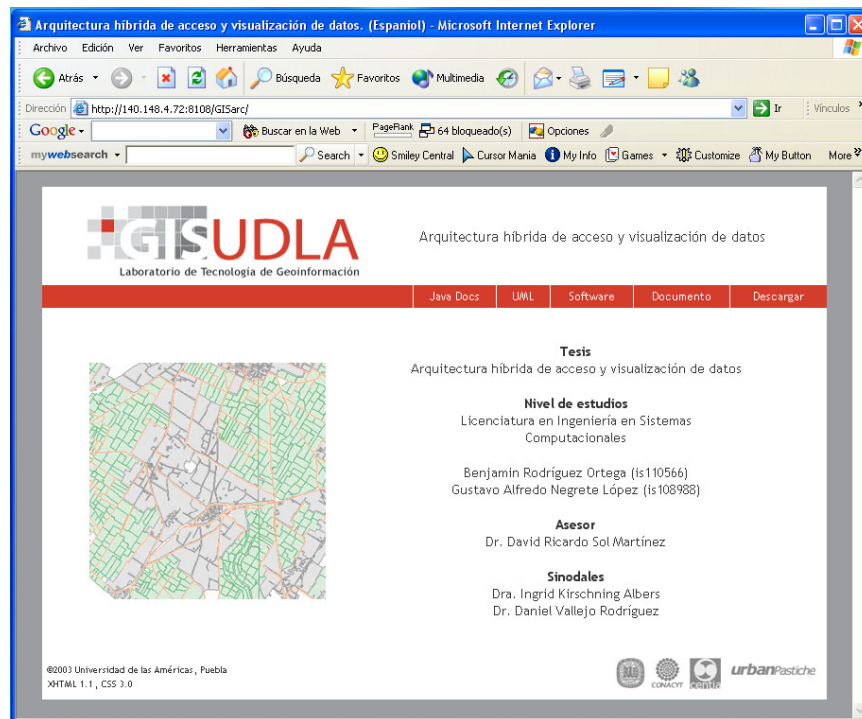


Figura 5.21. Pantalla inicial de la aplicación.

Una vez en la aplicación, aparecerán los documentos en GML y la descripción de las capas que vamos a visualizar. Si lo deseamos, podemos agregar una nueva capa, buscando el archivo y haciendo una actualización a la página. Tenemos la opción de escoger que capas y el orden en que serán desplegadas. Se cuenta con el botón de Borrar para limpiar todos los campos, así como con la opción de eliminar capas. Todo esto se muestra más detalle en la Figura 5.22.

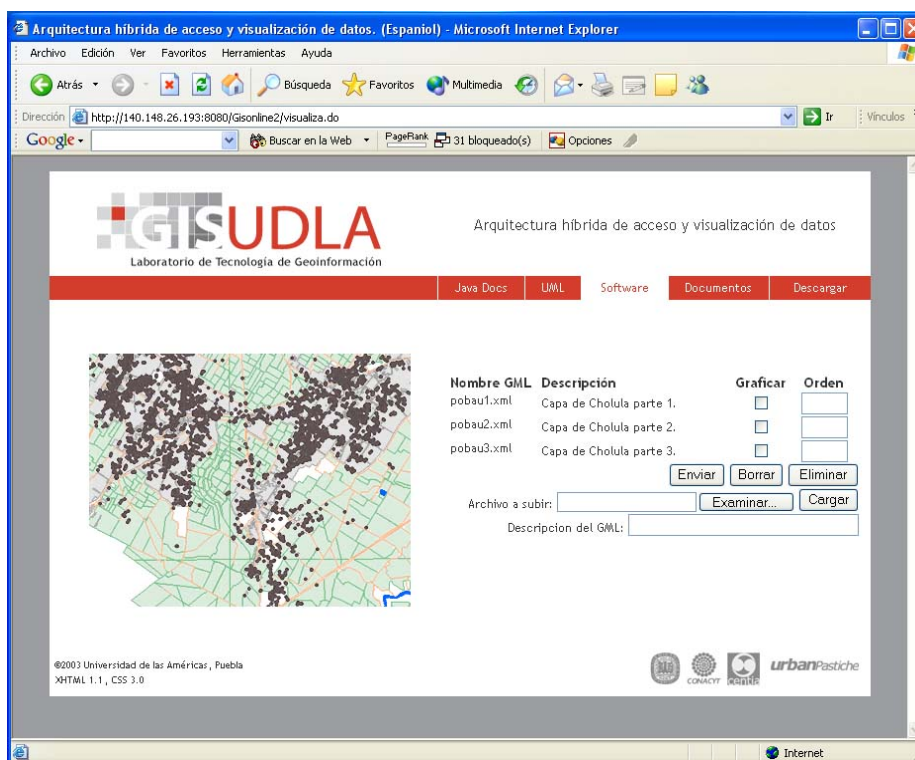


Figura 5.22. Funciones para visualizar capas.

Después de haber decidido las capas que se van a visualizar, así como su orden, y haber hecho clic en el botón de Enviar, aparecerán las capas a visualizar. Del lado izquierdo de la pantalla aparecerá el mapa con las distintas capas, y del lado derecho el mismo mapa pero en pequeño; debajo de éste los nombres de las capas que se están visualizando, así como un checkbox al lado de cada capa dando la opción de desplegarlas o no. Sobre el mapa ubicado del lado derecho aparecerá un cuadro, el cual de acuerdo al Zoom que se elija, va a marcar la parte del mapa que se está visualizando del lado izquierdo.

Se puede desplegar la información descriptiva de los features de las capas. Esto se logra haciendo clic sobre el feature en el mapa y nos despliega una ventana con sus atributos descriptivos.

La Figura 5.23 muestra un ejemplo de la visualización de las capas pobau1, pobau2 y pobau3 con un Zoom de 200%.

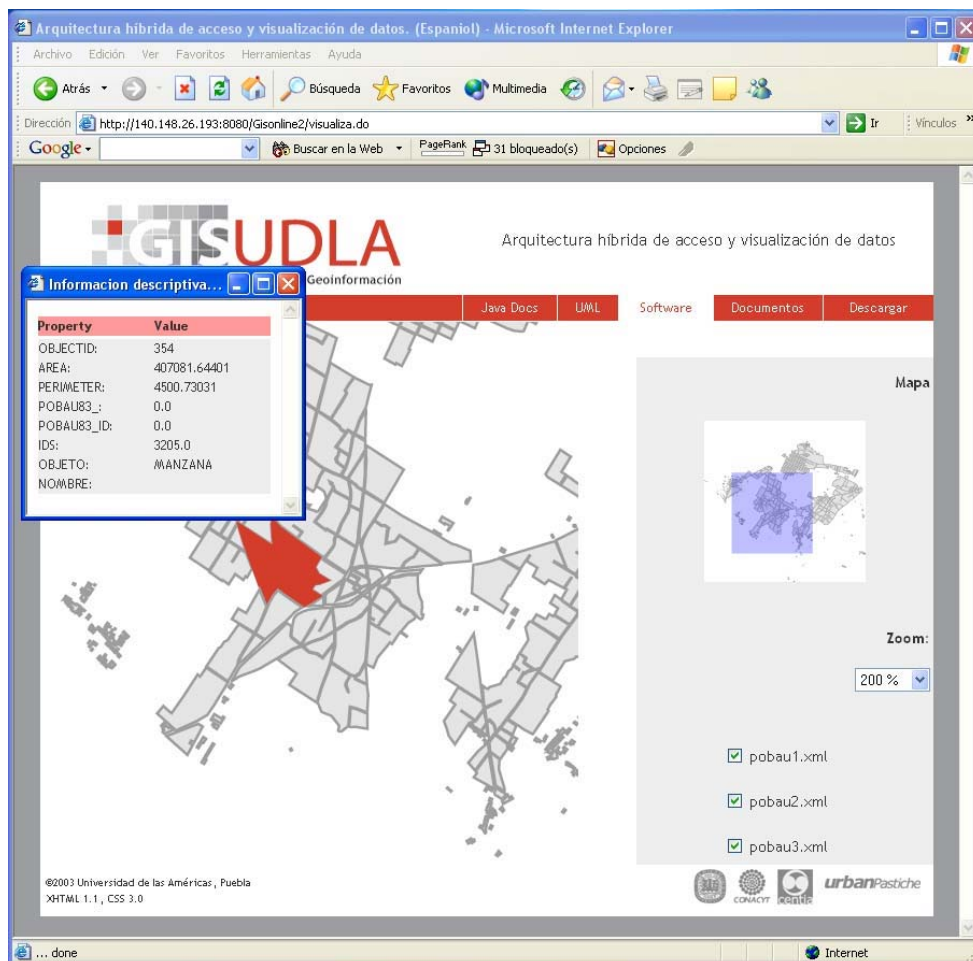


Figura 5.23. Visualización de un mapa.

5.10 Conclusión

En este capítulo se describió el desarrollo de nuestro trabajo, desde como iniciar el servicio de ArcSDE, editar las capas con ArcMap y ejecutar queries sobre ellas, para posteriormente almacenar los resultados en la base de datos mediante ArcCatalog y ArcSDE.

Una vez almacenadas las capas en la base de datos, se utilizó el trabajo desarrollado en el laboratorio por Gerardo Cepeda [Cepeda-2003]. En este trabajo se describe como se generaron los documentos GML, los cuales son traducidos a SVG con nuestra aplicación. También se mostró la interfaz de nuestra aplicación web de visualización en SVG, que cuenta con diversas opciones para desplegar las capas, como seleccionar las capas que se quieren ver, hacer Zoom sobre ellas y observar su información descriptiva.