



CAPÍTULO 2

MARCO TEÓRICO

En este capítulo se abordarán y describirán varios conceptos: los SIG (Sistemas de Información Geográfica o GIS por sus siglas en inglés); XML (eXtensible Markup Language), esquemas XML, Namespaces y GML (Geography Markup Language) una aplicación para datos geográficos basada en XML.

A continuación en la sección 2.1, se darán algunas definiciones de lo que es un SIG entre ellas la del Laboratorio de Tecnologías de Geoinformación de la UDLA (XALTAL). También se explicará su funcionalidad, su importancia, sus campos de aplicación y de como Internet ha sido pieza fundamental en la evolución de estos sistemas.

2.1 SISTEMAS DE INFORMACIÓN GEOGRÁFICA

El Sistema Corporativo de Información Geográfica de PEMEX (SICORI) define a un SIG como “una disciplina basada en conocimientos, metodologías y procedimientos asistidos por computadora que permiten la incorporación, almacenamiento, manipulación, procesamiento, consulta y presentación de información referenciada geográficamente en formatos gráficos y no gráficos” [SICORI, 2001]. El Environmental Systems Research Institute (ESRI) lo define como: “una herramienta computacional para trazar y analizar cosas que existan y sucesos que ocurren sobre la tierra” [ESRI, 2003].

En el laboratorio de Tecnologías de Geoinformación de la UDLA (XALTAL), pensamos que los SIG son una tecnología que provee mecanismos de adquisición, representación, almacenamiento, recuperación y visualización de datos con referencia espacial. Estos son muy útiles y permiten la construcción de aplicaciones para el apoyo a la toma de decisiones.



2.1.1 FUNCIONALIDAD DE UN SIG

Algunos argumentos básicos que fundamentan la utilización de un SIG [Domínguez, 2000] son:

- Permite gestionar un gran volumen de información a diferentes escalas y proyecciones.
- Realiza comparaciones entre escalas y perspectivas emulando una cierta capacidad de representación de diferentes lugares al mismo tiempo.
- Admite multiplicidad de aplicaciones y desarrollos; poniendo a nuestra disposición herramientas informáticas estandarizadas.
- Diferencia entre cambios cualitativos y cuantitativos; aportándonos una gran capacidad de cálculo.
- Integra espacialmente datos tabulares y geográficos junto a cálculos sobre variables espaciales o no espaciales (topología).

Los SIG están volviéndose cada vez más una herramienta importante casi para cualquier campo de aplicación debido a que puede asociar información diversa a un espacio geográfico determinado con el que está relacionado. El Instituto Nacional de Salud Pública (INSP) opina que "... un SIG introduce una nueva perspectiva que puede ayudar grandemente en la toma de decisiones y resolución de problemas" [INSP, 2003]. Si se desea profundizar más sobre el tema se recomienda consultar [Posada, 1999] y [Montiel, Méndez, 2002] donde se explican los conceptos de SIG ampliamente y se hace referencia a otros documentos y trabajos importantes.

2.1.2 CAMPOS DE APLICACIÓN DE UN SIG

Los SIG son útiles en los negocios, el gobierno y en la investigación. Por la parte de los negocios se usan en tareas como estudios de factibilidad para el establecimiento de nuevas sucursales según factores demográficos, vías de acceso, niveles socioeconómicos de la población y existencia de competidores. También se utilizan para redefinir los territorios de ventas, en la búsqueda de menores gastos de transportación, de almacenamiento y en menores tiempos de operaciones, que resulta en la posibilidad de dedicar mayor atención en la formulación de estrategias para captar clientes y mejorar los productos o servicios.



De igual forma, los SIG son herramientas importantes para la gestión del catastro, el desarrollo urbano, cuantificar y tener información actualizada sobre los recursos naturales de un país, el registro público, instalaciones eléctricas, de agua, gas y comunicaciones, así como el seguimiento de transportes en tiempo real. También se utiliza en la búsqueda de las mejores rutas de distribución. Son útiles para dar respuesta a emergencias, en seguridad pública, en inventarios arqueológicos, en identificación de áreas de riesgo, en reservas ecológicas y recursos naturales; así como en agricultura y nuevos asentamientos. Igualmente se usan para elaborar mapas con fines de mercadotecnia, turismo, difusión, educación y seguramente otras áreas, porque difícilmente habrá tareas de administración, planeación y estudio que no den mejores resultados con la ayuda de un SIG.

2.1.3 LOS SIG EN INTERNET

[Harder, 1998] opina que internet no cambia la naturaleza básica de los SIG, la pone "Online". Los servidores de mapas tienen su primer desarrollo en 1993 con el Xerox PARC Map Viewer. Los avances más recientes se han enfocado hacia las estructuras cliente-servidor, SIG's basados en la tecnología orientada a objetos y SIG distribuidos, mejor conocidos como sistemas de IGD's (Información Geográfica Distribuida) [Montiel, Méndez, 2002].

Se denomina Información Geográfica Distribuida (IGD) a los productos o servicios que se relacionan con el empleo de las tecnologías de Internet, las cuales permiten a los usuarios tener acceso a la información geográfica distribuida en una gran variedad de formas como lo pueden ser mapas, imágenes, conjuntos de datos, análisis de información y reportes [Plewe, 1997].

[Domínguez, 2000] hace la siguiente clasificación para los SIG Distribuidos o IGD's:

- Los más simples son las aplicaciones estáticas, es decir, el servidor web pone a disposición del usuario una imagen en formato GIF o JPG. Estos mapas sólo muestran localizaciones. Ejemplos los encontramos en el sitio de de MapQuest [MapQuest, 2003] en la sección de mapas (ver figura 2.1).
- Una variación de estas aplicaciones serían aquellas que se actualizan cada determinado tiempo. Un ejemplo de esto sería la página del CENAPRED



[CENAPRED, 2003] en la sección de monitoreo del volcán Popocatepetl donde la imagen del volcán se actualiza cada minuto.

- Aplicaciones un poco más complejas son las de los SIG distribuidos, que a partir de peticiones de un usuario generan un mapa como respuesta. En estas aplicaciones el usuario genera su propio mapa. Un ejemplo de esto es el servicio que ofrece el Instituto de Estadística de Andalucía [IEA, 2003] que a partir de una variable dada por el usuario genera mapas de provincias de esta comunidad autónoma.

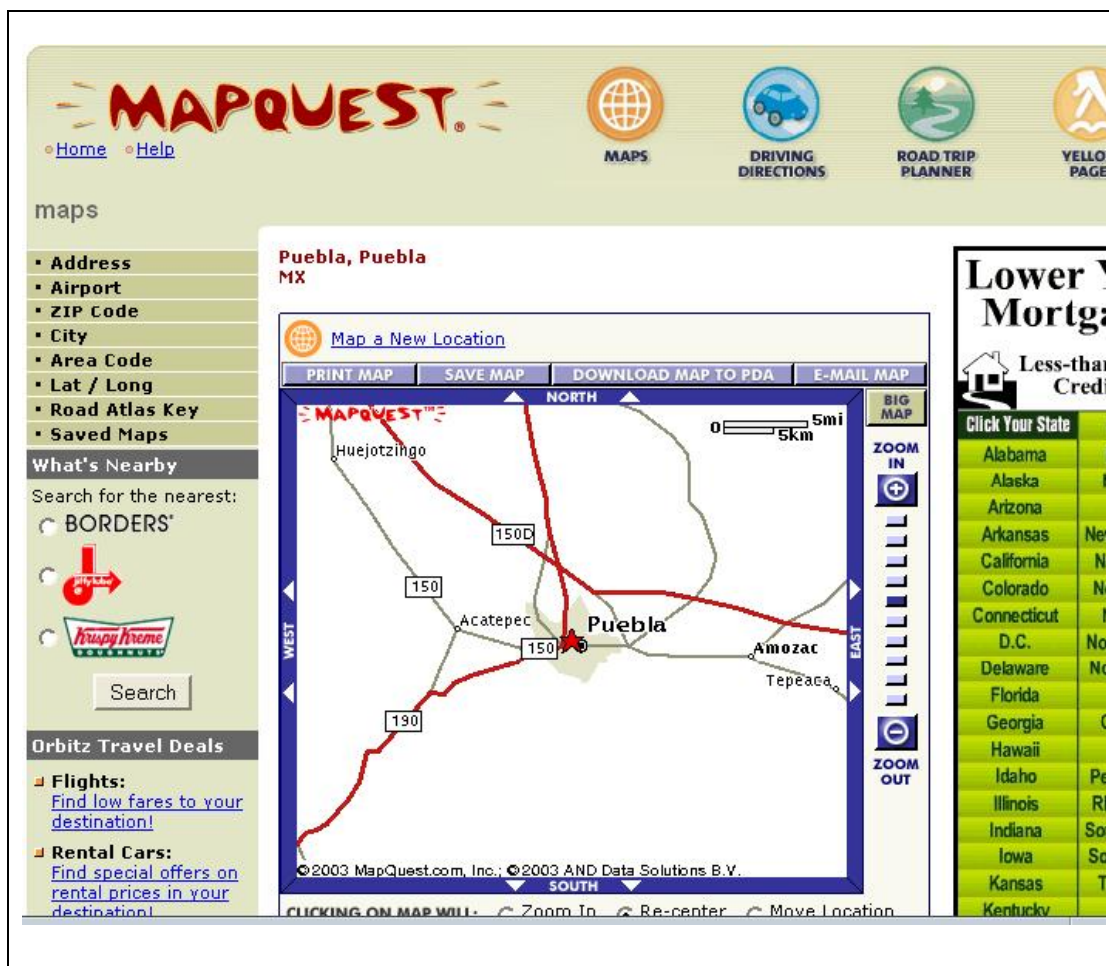


Figura 2.1 Mapa del estado de Puebla (fragmento)

- El siguiente grupo lo conforman los sitios en los cuales el resultado es un mapa producto de un análisis espacial como pueden ser búsquedas geográficas y búsquedas condicionales. Ejemplo de esto es el servicio "ATM Locator" del sitio



de VISA el cual ayuda a encontrar todos los cajeros automáticos existentes en una zona dada [VISA, 2003].

- Otra opción se presenta con el resultado de mapas generados a partir de un procesamiento de datos geográficos, en el que el servidor presenta los datos almacenados como respuesta a una petición del usuario. Ejemplo de esto es el servicio de mapas del atlas en línea de la National Geographic [NationalGeographic, 2003], (ver figura 2.2).
- Los servicios de datos públicos. En este caso un organismo pone a disposición del público datos geográficos para que puedan ser descargados en su computadora. Ejemplo de este servicio es el que ofrece el Instituto Nacional de Estadística Geográfica en Informática (INEGI) que pone a disposición del público información geográfica y estadística diversa sobre México [INEGI, 2003].



Figura 2.2 Servicio de Atlas en línea de National Geographic



- Los servicios de venta de datos. El objetivo de estos es el mismo que el anterior pero con fines de lucro.

A este contexto se le suma el surgimiento de la Tecnología XML (eXtensible Markup Language) - en ocasiones llamado el ASCII de Internet -. Sin duda una tecnología de vanguardia que ha revolucionado el intercambio de información enormemente y el cual es de gran importancia en este proyecto de tesis.

2.2 EL ESTÁNDAR XML

El intercambio de información siempre ha sido un grave problema cuando se utilizan lenguajes y sistemas operativos heterogéneos. Este problema sólo se agudizó con la llegada de Internet, inclusive previo a su aparición fueron creados mecanismos para lograr el intercambio fluido de información entre diferentes sistemas. El primer método fue GML (General Markup Language), posteriormente SGML (Standard Generalized Markup Language) y actualmente XML. Estos mecanismos son llamados lenguajes de marcado "markup" o meta-lenguajes [Rubio, 2002].

El grupo responsable del desarrollo de éste estándar es el XML Working Group, el cual fue promovido por el W3C (World Wide Web Consortium) en el año de 1996 y fue presidido por Jon Bosak de SUN Microsystems. La versión 1.0 de XML es una recomendación del W3C desde el 10 de febrero de 1998 [Bosak, 2000]. Esta especificación es un subconjunto del estándar SGML que data de 1986 definido por ISO 8879 y desarrollado en un principio por IBM hace tres décadas.

2.2.1 DEFINIENDO XML

[Ray, 2001] define XML de acuerdo a 3 niveles:

- Primer Nivel: Un protocolo que nos permite tener acceso al contenido y manejo de la información. La mayoría de los usuarios se limitan al uso de este nivel.
- Segundo Nivel: Una familia de tecnologías que pueden hacerlo todo, desde dar formato a un documento hasta hacer filtraciones de datos.



- Tercer Nivel: Una filosofía para el manejo de información que busca la máxima utilidad y flexibilidad de los datos, refinándolos a su forma más pura y estructurada.

Cabe señalar que la finalidad de XML no es la de ser un documento final. Se necesitan otros recursos para darle presentación al documento. Tales recursos pueden ser las hojas de estilo (*stylesheets*), plantillas de traducción XSLT o alguna aplicación en general. En la figura 2.3 se muestra un ejemplo de un documento en XML.

Todo documento XML puede ser representado en forma de árbol. Cada parte del documento ya sea elemento, atributo, texto o comentario se representa como un nodo [Ray, 2001].

```
<?xml version="1.0" encoding="UTF-8"?>
<CD>
  <titulo>Tracy Chapman</titulo>
  <artista>Tracy Chapman</artista>
  <año>1988</año>
  <disquera>Elektra</disquera>
  <canciones>
    <track1>Fast car</track1>
    <track2>Across the lines</track2>
  </canciones>
</CD>
```

Figura 2.3 Ejemplo de un documento XML

En la figura 2.4 se muestra un diagrama de árbol que representa el documento "CD". El nodo o elemento "CD" es el elemento raíz, debido a que encapsula a todos los demás elementos. Los elementos con línea más delgada son los elementos o nodos hijos de CD y los elementos con líneas punteadas son nodos hijos del elemento canciones.

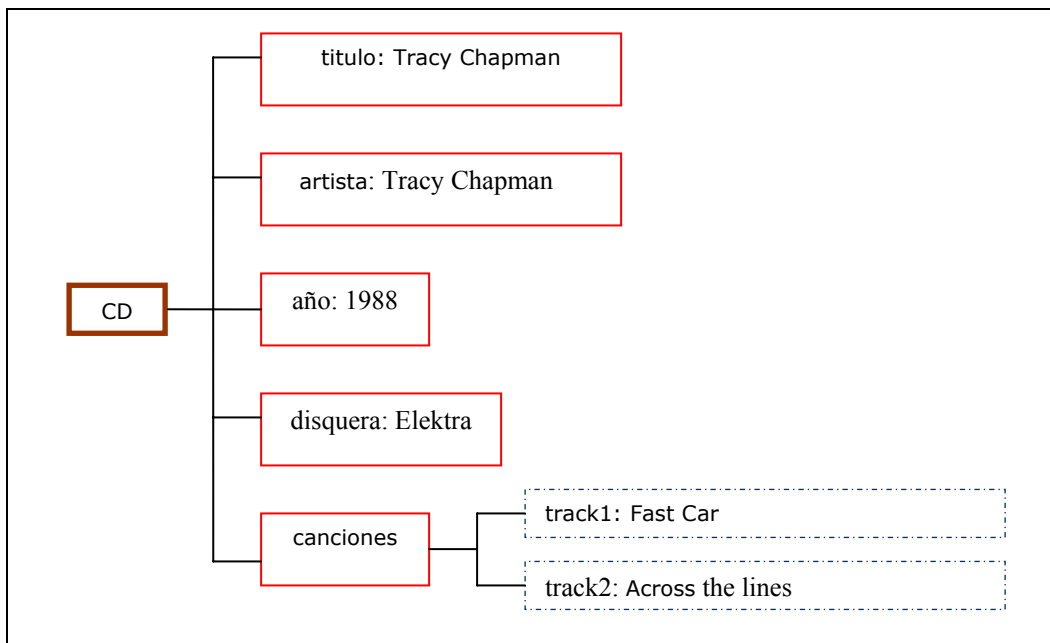


Figura 2.4 Diagrama de árbol del documento XML de ejemplo

2.2.2 DOCUMENTOS XML BIEN FORMADOS Y VÁLIDOS

La especificación oficial de XML se puede encontrar en el sitio [Bray, Paoli, 1999]. Los documentos XML son considerados bien formados si cumplen con las reglas sintácticas de esta especificación. Si se desea crear un nuevo lenguaje de marcado, una aplicación en específico y/o se desea que el documento XML siga una estructura y gramática determinada es necesario validarlos a partir una definición de documento hecha a partir de un DTD (Document Type Definition) o un esquema XML.

Un DTD a grandes rasgos es una definición de los elementos que puede haber en el documento XML, y su relación entre ellos, atributos y posibles valores. Sin embargo tiene varias deficiencias. La primera y más obvia es su falta de expresividad, pues está escrito en formato Extended Backus Naur Form [Rubio, 2002], un formato totalmente ajeno a XML, donde no pueden definirse restricciones de tipo (string, int, entre otros).

El propósito de un esquema o XML Schema (al igual que un DTD) es definir una clase de documentos XML, por lo tanto el término de "documento instancia" se usa a menudo para describir un documento que conforma con un esquema en particular [Fallside, 2001]. El concepto de esquema surge como solución a las deficiencias de los



DTD's. Este se basa fuertemente en XML, es más descriptivo, permite la definición de tipos y utiliza ampliamente el concepto Namespace, el cual no es soportado en su totalidad por los DTD's. El concepto de Namespace surge de la necesidad de combinar diferentes vocabularios (definiciones de tipos de datos) en XML. Es una forma de agrupar varios elementos para ser usados en un mismo documento [Rubio, 2002].

Un XML Namespace es un conjunto de nombres identificados por una referencia URI (Uniform Resource Identifier), que se utilizan en documentos XML como tipos de elemento y nombres de atributo. Los Namespaces o espacios de nombres de XML proporcionan un método simple para calificar nombres de elementos y atributos usados para documentos XML, asociándolos con otros Namespaces identificados por referencias URI [Bray, 1999].

Una de las razones por las que surgen los Namespaces es la modularidad. Si existe un vocabulario bien entendido y para el cual hay programas útiles disponibles, es mejor reutilizar este vocabulario y no inventarlo. Otra buena razón por la que fue necesario que surgieran los Namespaces son los documentos normados por varios esquemas. Estos documentos que contienen varios vocabularios de formato, plantean problemas de reconocimiento y colisión. Los módulos de software tienen que ser capaces de reconocer las etiquetas y atributos para cuyo procesamiento fueron diseñados, incluso si se producen colisiones cuando un código de formato destinado a algún otro paquete de software usa el mismo nombre de atributo o tipo de elemento. Esto exige que las estructuras de los documentos deban tener nombres universales, cuyo ámbito de aplicación se extienda más allá del documento que las contiene [Bray, 1999].

Más adelante en la sección 2.4.3 se hablará un poco más sobre las ventajas y funcionalidades del concepto de esquemas XML. Si se desea profundizar sobre aspectos técnicos de XML se recomienda consultar [Ray, 2001], [Montiel, Méndes, 2002] y [Barbero, 1999]. Si se desea profundizar más sobre aspectos técnicos de Namespaces se recomienda consultar [Bray, 1999].



2.2.3 VENTAJAS DE XML

Debido a su gran flexibilidad el estándar XML se puede usar para un infinidad de trabajos y aportar muchas ventajas en varios campos prácticos [Montiel, Méndez, 2002]. Algunas ventajas de XML en algunas aplicaciones son:

- **Migración de datos:** Un caso particular es la migración de datos entre diferentes Bases de datos. Hacerla usando XML sería más sencillo.
- **Presentación de la información diversa:** Con XML podemos generar varias presentaciones de la información de un solo archivo con el uso de hojas de estilo.
- **Filtrado de la información:** Se puede hacer también un filtro de la información de interés dependiendo de cada usuario por medio de plantillas de Traducción XSLT. También por medio de éstas o de alguna otra aplicación se puede generar información en cualquier otro formato a partir de un documento XML. Un ejemplo de esto es la generación de mapas con SVG a partir de un documento GML (Geography Markup Language).
- **Nuevos lenguajes de Marcado:** XML nos da la posibilidad de definir nuestros propios lenguajes para procesar información de una aplicación en específico.

A continuación veremos el lenguaje de marcado GML (Geography Markup Language). Dicho lenguaje es de sumo interés para esta tesis y será descrito en la siguiente sección.

2.3 GEOGRAPHY MARKUP LANGUAGE

GML (Geography Markup Language) es una especificación XML para el transporte y almacenamiento de información geográfica, incluyendo las propiedades espaciales y no espaciales de los rasgos geográficos, también llamados "features". Esta especificación está siendo desarrollada por el OpenGis Consortium (OGC) [Cox, 2002]. Está respaldado por una gran variedad de empresas incluyendo Oracle Corporation, Galdos Systems Inc., MapInfo, cubeWerx y compusult Ltd, entre otros. GML fue implementado y probado con éxito a través de una serie de demostraciones que



forman parte del Web Mapping Test Bed (WMT) del OpenGis Consortium en Septiembre de 1999.

“Interoperabilidad es la base de esta actividad de GML” dice Roger Harwell, director de soluciones SIG para GeoMedia WebMap products en Inergraph Corp. “Esto permitirá interoperabilidad entre varios vendedores, así que los usuarios podrán tomar datos de cualquier lugar que ellos quieran para darle un uso. Esto solo hace los datos mucho más accesibles a la gente” [Robinson, 2001].

“Pienso que GML tiene un rol grande que jugar en el gobierno, particularmente con servicios basados en localización”, dice Joe Sewash, analista SIG en el Departamento de Finanzas y Oficina de Administración para Recursos Informáticos de Tennessee. “Si se tiene una agencia con mucho trabajo de campo, por ejemplo, existe una tendencia a tener muchas bases de datos diferentes con links SIG explícitos a ellas. GML puede ser usado para ligar muchos datos no espaciales tradicionales con datos espaciales” [Robinson, 2001].

Efectivamente, el uso colectivo de esta especificación permite incluir los datos geográficos, en flujos automatizados (sindicación geográfica y bases de datos distribuidas). También a partir de este formato se puede proveer de archivos en diferentes formatos (imágenes, PDF, Postscript, SVG, VRML, o formatos comerciales como SHP) y mostrarse en una aplicación o un Navegador Web estándar.

2.3.1 DATOS GEOGRÁFICOS Y MAPAS

Antes de pasar a GML en detalle se debe puntualizar algunos conceptos, el primero es la diferencia entre dato geográfico (que esta codificado en GML), y las interpretaciones que se den a ese dato, ya sea mostrándolo en un mapa o cualquier otra forma de visualización. Un dato geográfico es una abstracción del mundo en términos de las propiedades y geometrías que lo caracterizan independientemente de cualquier visualización particular de ese dato. Así como XML está ayudando a la Web a separar claramente el contenido de la presentación, GML hace lo mismo en el mundo de la geografía [Lake, 2000].



2.3.2 GML ES TEXTO

El segundo punto a tratar es que GML así como cualquier codificación XML, representa información geográfica en la forma de texto. Esto agrega simplicidad y visibilidad de su lado. Es muy fácil de examinar y manipular [Lake, 2000].

2.3.3 ESTRUCTURA DE GML

GML está basado en un modelo abstracto de geografía desarrollado por el OGC. Este describe el mundo en términos de entidades geográficas llamadas "features". GML en su segunda versión introduce tres esquemas para validar la estructura de los documentos, estos son *geometry*, *feature* y *xlinks*; los dos primeros sustituyen el DTD de GML 1.0. El esquema XLink provee los atributos de XLink para soportar funcionalidades de ligado.

Xlink describe una manera estándar de añadir hiper-enlaces a un archivo XML. Es un lenguaje, definido en términos de marcas (tags) XML, que nos permite introducir enlaces en nuestros documentos XML, de modo que podamos relacionar unos documentos con otros. Un XLink permite establecer una relación entre dos o más recursos en la Web, sin que necesariamente estos recursos sepan que están enlazados [Goldberg, Prescod, 2000]. Si se desea profundizar más sobre la tecnología XLink se recomienda consultar [DeRose, 2001].

El uso de esquemas XML añade un conjunto rico de tipos de datos primitivos como string, boolean, float y month, entre otros. También permite la creación de tipos de datos definidos por el usuario. XML ofrece varias ventajas al usarse para restringir codificaciones de GML:

- Permite la mezcla entre diferentes vocabularios usando "Namespaces".
- Permite un mejor control sobre la estructura de la jerarquía de definición de tipos.
- Otorga extensibilidad y flexibilidad por medio de tipos derivados y grupos de substitución.



Los esquemas base de GML efectivamente proveen un meta-esquema, del cual puede ser construido un esquema de aplicación. A continuación se explicará de manera muy simple las directivas del esquema de geometrías y features, sin profundizar en XLink. Si se desea profundizar más sobre la tecnología de esquemas XML se recomienda consultar [Fallside, 2001], [Thompson, 2001], [Biron, Ashok, 2001].

2.3.3.1 FEATURES (RASGOS GEOGRÁFICOS)

La OGC define a un *feature* geográfico como la abstracción de un fenómeno real que está asociado a una ubicación relativa con la Tierra (Spatial Reference System). Un *feature* geográfico es aquel cuyas propiedades pueden ser estimadas geoméricamente y el número de propiedades que puede tener un *feature* está determinado por su definición de tipo (esto se determina en el esquema de aplicación que lo norma).

Un *feature* geográfico se compone de propiedades descriptivas y geométricas. Los datos descriptivos están compuestos por tuplas del tipo (nombre, tipo, valor). GML define por defecto los datos descriptivos opcionales: **descripción**, **nombre**, **fid** (un tipo ID único para cada *feature*) y **boundedBy** atributo de tipo GML *Box* usado para definir los límites de un *feature* dado. Los datos geométricos están compuestos por figuras básicas como punto, línea, polígono y multigeometrías. GML permite también definir geometrías propias en caso de ser necesario.

2.3.3.2 FEATURE COLLECTIONS (COLECCIONES DE FEATURES GEOGRÁFICOS)

GML 2.1.2 soporta la construcción de *Feature Collections*. Este elemento como su nombre lo indica es una colección de *features*, que puede ser considerado así mismo como un *feature* y puede contener incluso a otras colecciones de *features*. Cada uno debe usar la propiedad **featureMember** para cada *feature* o *feature collection* contenido. La especificación obliga al uso del elemento **boundedBy** y al ser considerado también como un *feature* puede tener propiedades propias además de las propiedades de los *features* que contiene.



2.3.3.3 CODIFICANDO GEOMETRÍAS

GML define dos maneras para representar las coordenadas de un elemento geométrico. La primera es por medio de etiquetas `<coord>` (ver figura. 2.5) que encapsulan componentes de tuplas X,Y,Z y la segunda es por medio de una sola cadena contenida entre etiquetas `<coordinates>` cada coordenada X,Y,Z va separada por coma y cada tupla se separa por un espacio en blanco o salto del línea (ver figura. 2.6) La ventaja de usar etiquetas `<coord>` es que facilita el análisis del código GML. En esta versión se introduce el manejo de coordenadas para una, dos o tres dimensiones.

Cabe señalar que aunque esta versión de GML permite que se especifiquen coordenadas en tres dimensiones, no da un soporte directo para construir geometrías tridimensionales.

```
<gml:coord>
  <gml:X>100.0</gml:X>
  <gml:Y>50.0</gml:Y>
</gml:coord>
```

Figura 2.5 Sintaxis de Coords.

Las coordenadas de una geometría son definidas dentro de algún Spatial Reference System (SRS), y todas las geometrías deben especificar su SRS. El atributo **srsName** puede ser usado para comprobar la equivalencia de SRS entre diferentes geometrías. También hay un atributo opcional *gid* de tipo ID, que representa un identificador único para cada geometría.

```
<gml:coordinates>
  100.0,50.0
  45.343,456.12
</gml:coordinates>
```

Figura 2.6 Sintaxis de Coordinates.

A continuación se explicará de manera sencilla como se representan las geometrías en esta versión de GML:



- **Box:** Este elemento se usa para codificar límites. Cada elemento *Box* contiene una secuencia de dos coordenadas. Estas representan los valores mínimos y máximos del límite (ver figura 2.7).

```
<gml:Box srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
  <gml:coord>
    <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y> ← X,Y min
  </gml:coord>
  <gml:coord>
    <gml:X>100.0</gml:X> <gml:Y>100.0</gml:Y> ← X,Y max
  </gml:coord>
</gml:Box>
```

Figura 2.7 Sintaxis de Box

- **Point:** Este elemento se usa para codificar un punto el cual sólo puede contener un solo elemento de coordenadas (ver figura 2.8).

```
<gml:Point srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
  <gml:coord>
    <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
  </gml:coord>
</gml:Point>
```

Figura 2.8 Sintaxis de Point

- **LineString:** Este elemento se usa para codificar una secuencia de líneas rectas, una trayectoria cerrada se indica haciendo coincidentes la primera y última coordenadas, este último coincide con la definición de *LinearRing* (ver figura. 2.9).

```
<gml:LineString srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
  <gml:coord>
    <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
  </gml:coord>
  <gml:coord>
    <gml:X>20.0</gml:X> <gml:Y>35.0</gml:Y>
  </gml:coord>
  <gml:coord>
    <gml:X>100.0</gml:X> <gml:Y>100.0</gml:Y>
  </gml:coord>
</gml:LineString>
```

Figura 2.9 Sintaxis de LineString



- **LinearRing:** Este elemento se usa para codificar una secuencia cerrada de líneas rectas. Deben coincidir la primera y la última coordenadas. Como se usa para la construcción de Polígonos no se requiere del atributo **srsName** (ver figura 2.10).

```
<gml:LinearRing>
  <gml:coord>
    <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
  </gml:coord>
  <gml:coord>
    <gml:X>20.0</gml:X> <gml:Y>35.0</gml:Y>
  </gml:coord>
  <gml:coord>
    <gml:X>100.0</gml:X> <gml:Y>100.0</gml:Y>
  </gml:coord>
  <gml:coord>
    <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
  </gml:coord>
</gml:LinearRing>
```

Figura 2.10 Sintaxis de LinearRing

- **Polygon:** Este elemento se usa para codificar polígonos compuestos por un conjunto de *LinearRings*. Un Polígono debe tener un límite exterior (contenido dentro de un elemento *<outerBoundaryIs>*) y cero o varios límites interiores (cada uno contenido dentro de un elemento *<innerBoundaryIs>*, ver figura 2.11)

```
<gml:Polygon srsName="http://www.opengis.net/gml/srs/epsg.xml#4326">
  <gml:outerBoundaryIs>
    <gml:LinearRing>
      <gml:coord> <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
      </gml:coord>
      <gml:coord> <gml:X>20.0</gml:X> <gml:Y>35.0</gml:Y>
      </gml:coord>
      <gml:coord> <gml:X>100.0</gml:X> <gml:Y>100.0</gml:Y>
      </gml:coord>
      <gml:coord> <gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y>
      </gml:coord>
    </gml:LinearRing>
  </gml:outerBoundaryIs>
</gml:Polygon>
```

Figura 2.11 Sintaxis de Polygon

Esta especificación también define varias colecciones geométricas, todas estas colecciones usan una propiedad de pertenencia para los elementos que contienen.



Cabe señalar que la geometría que contiene a las demás es la única que debe utilizar el atributo **srsName**. A continuación se verán estas colecciones:

- **MultiGeometry:** Esta es una colección heterogénea de geometrías y puede contener cualquier geometría básica y a otras colecciones geométricas. Su elemento de pertenencia es un *geometryMember* (ver figura 2.12).

```
<gml:MultiGeometry srsName="http://www.opengis.net/gml/epsg.xml#4326">
  <gml:geometryMember>
    <gml:Point>
      <gml:coord> <gml:X>0.0</gml:X> <gml:Y>50.0</gml:Y></gml:coord>
    </gml:Point>
  </gml:geometryMember>
  <gml:geometryMember>
    <gml:LineString>
      <gml:coord><gml:X>0.0</gml:X> <gml:Y>0.0</gml:Y></gml:coord>
      <gml:coord><gml:X>20.0</gml:X> <gml:Y>35.0</gml:Y></gml:coord>
    </gml:LineString>
  </gml:geometryMember>
</gml:MultiPoint>
```

Figura 2.12 Sintaxis de MultiGeometry

- **MultiPoint:** Esta es una colección de puntos. Su elemento de pertenencia es un *pointMember* (ver figura 2.13).
- **MultiLineString:** Esta es una colección de *lineStrings*. Su elemento de pertenencia es un *lineStringMember*.
- **MultiPolygon:** Esta es una colección de polígonos. Su elemento de pertenencia es un *polygonMember*.

```
<gml:MultiPoint srsName="http://www.opengis.net/gml/epsg.xml#4326">
  <gml:pointMember>
    <gml:Point>
      <gml:coord> <gml:X>0.0</gml:X> <gml:Y>50.0</gml:Y>
    </gml:Point>
  </gml:pointMember>
  <gml:pointMember>
    <gml:Point>
      <gml:coord> <gml:X>70.0</gml:X><gml:Y>60.0</gml:Y>
    </gml:Point>
  </gml:pointMember>
  <gml:pointMember>
    <gml:Point>
      <gml:coord><gml:X>100.0</gml:X><gml:Y>50.0</gml:Y>
    </gml:Point>
  </gml:pointMember>
</gml:MultiPoint>
```

Figura 2.13 Sintaxis de MultiPoint



2.3.3.4 CODIFICANDO FEATURES CON GEOMETRÍAS

A continuación se muestra el siguiente ejemplo usado para modelar los departamentos de cada carrera en la universidad (ver figura 2.14).

```
<Departamento>
  <nombre>Departamento de Sistemas</nombre>
  <jefeDepartamento>David Sol</jefeDepartamento>
  <gml:PointProperty>
    <gml:Point>
      <gml:coord>
        <gml:X>1.0</gml:X>
        <gml:Y>1.0</gml:Y>
      </gml:coord>
    </gml:Point>
  </gml:PointProperty>
</Departamento>
```

Figura 2.14 Fragmento en GML de Departamento

Este ejemplo está basado en el siguiente fragmento de esquema de aplicación (ver figura 2.15).

```
<xs:element name="Departamento" type="ex:DepartamentoType"
  substitutionGroup="gml:Feature"/>
<xs:complexType name="DepartamentoType">
  <xs:complexContent>
    <xs:extension base="gml:AbstractFeatureType">
      <xs:sequence>
        <xs:element name="nombre" type="xs:string"/>
        <xs:element name="jefeDepartamento" type="xs:string"/>
        <xs:element ref="gml:pointProperty"/>
        <!-- Punto que apunta a la ubicación del departamento -->
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

Figura 2.15 Fragmento del esquema XML de Departamento

En este esquema se definieron dos datos descriptivos: **nombre** y **jefeDepartamento**, ambos de tipo string y un dato geométrico de tipo *pointProperty* que es un punto que apunta a la ubicación del departamento.

2.3.3.5 GML HOY

GML favorece ampliamente la interoperabilidad, el intercambio y manejo de datos, permitiendo un mayor control sobre ellos. Al ser un estándar permite que



prácticamente cualquier persona pueda crear, manipular, distribuir y consultar gran cantidad de datos geográficos, revolucionando el mundo GIS.

El Servicio Geográfico del Reino Unido adoptó recientemente GML como un medio para compartir todos sus datos geográficos en toda la gran Bretaña. GML también se ha convertido en la clave para la estandarización del desarrollo de Infraestructuras Geo-Espaciales de Canadá. Si se desea profundizar más sobre GML2.1.2 se recomienda consultar [Cox, 2002].

En este capítulo se trataron los principales conceptos de esta tesis, empezando por una breve introducción de lo que son los sistemas de información geográfica y su gran utilidad, pasando por XML, Esquemas y Namespaces, para finalmente dar entrada a lo que es GML parte fundamental de todo este proyecto. Para manejar los documentos XML es necesario el uso de herramientas que nos faciliten de manera práctica y sencilla estas labores. En la primera sección del capítulo 4 se mencionará y analizarán las principales tecnologías para la creación y manipulación de XML. Antes se describirá brevemente algunos proyectos relacionados con este proyecto de tesis.