

79



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

ANALISIS Y DISEÑO DE UN SISTEMA DE INFORMACION GEOGRAFICA PARA ANALISIS BIOGEOGRAFICOS.

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERIA EN COMPUTACION

P R E S E N T A :
FIDENCIO ROJAS SANDOVAL

DIRECTOR DE TESIS: SANTIAGO IGOR VALIENTE GOMEZ



CIUDAD UNIVERSITARIA, MEXICO, D.F.

2000

289336



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

**Universidad Nacional Autónoma de
México.
Facultad de Ingeniería.**

**Análisis y Diseño de un Sistema de Información
Geográfica para Análisis Biogeográficos.**

Tesis

**Que presenta para obtener el título de
INGENIERÍA EN COMPUTACIÓN.**

Presenta:

Fidencio Rojas Sandoval.

**Director de tesis : Santiago Igor Valiente
Gómez.**

Ciudad Universitaria, D.F. 2000

Dedicatorias:

A los mejores y más queridos ejemplos de vida: mis Padres.

A mis hermanas y hermanos, por el maravilloso regalo que me han dado: su compañía y apoyo.

A Alejandra, por que he encontrado lo que me hace falta y quiero compartir todo contigo, Te Amo.

A Miguel, porque este triunfo también es tuyo y eres para mi uno de mis mejores amigos.

A Alejandro y Víctor por brindarme lo mejor de ustedes:
Su Amistad

Agredecimientos:

A la Facultad de Ingeniería de la Universidad Nacional Autónoma de México, por brindarme la preparación profesional para ser mejor Mexicano.

Al Ing. Santiago Igor Valiente Gómez, por todo lo que me ayudo para guiarme en este camino.

A mis Sinodales:

Ing. Alberto Templos Carvajal, M. de I. Adolfo Millan Najera, Ing. Jaquelina López Barrientos y a la Ing. Laura Sandoval Montaña por su ayuda en la realización de este trabajo.

A Urban Science de México, por el tiempo concedido para la culminación de mi meta.

Y a todos aquellos, que me han brindado su apoyo y me han brindado la oportunidad de conocerlos....



Análisis y Diseño de un Sistema de Información Geográfica para Análisis Biogeográficos

1	Introducción.....	1
1.1	Objetivo:.....	1
1.2	Justificación :.....	1
1.3	Planteamiento del Problema:.....	2
2	Sistemas de Información Geográfica.....	4
2.1	¿Qué es un Sistema de Información Geográfica (SIG)?.....	4
2.2	Conceptos de SIG.....	6
2.2.1	Geodesia.....	6
2.2.2	Cartografía.....	6
2.2.3	Cartografía por Computadora.....	7
2.2.4	Mapas.....	10
2.2.5	Tipos de SIG.....	15
2.2.6	Antecedentes de SIG.....	16
2.2.7	Estructura de un SIG.....	17
2.3	Alcances y Usos de los SIG.....	30
2.4	Sistemas de SIG Existentes.....	32
3	Área de Estudio: Biogeografía.....	34
3.1	Biogeografía.....	34
3.2	Biogeografía Cuantitativa.....	37
3.2.1	Especie.....	37
3.2.2	Definición del Área de Estudio.....	39

3.2.3	Definición OGU's.....	39
3.2.4	Matriz de Presencia-Ausencia.....	40
3.2.5	Índices de Diversidad.....	40
3.2.6	Estimadores de Riqueza.....	40
3.2.7	Distribución Geográfica y Lista de Especies.....	41
3.2.8	Similitud Biogeográfica.....	42
3.2.9	Matriz de Similitud.....	42
3.2.10	Análisis de Agrupamiento.....	42
3.2.11	Definición de Áreas Prioritarias para la Conservación.....	45
3.2.12	Jerarquía y Escala.....	45
3.2.13	El Problema de la Unidad de Área Modificable.....	47
3.3	Algunas Consideraciones:.....	48
4	<i>Discusión de la Metodología a Seguir.....</i>	49
4.1	Ciclo de Vida de un Sistema de Información (SI).....	50
4.1.1	Cascada Pura.....	51
4.1.2	Codificar y Corregir.....	53
4.1.3	Espiral.....	53
4.1.4	Cascadas Modificadas.....	55
4.1.5	Prototipo Evolutivo.....	56
4.1.6	Entregas por Etapas.....	56
4.1.7	Diseño por Planificación.....	57
4.1.8	Entrega Evolutiva.....	58
4.2	Paradigmas de Programación.....	58
4.2.1	Programación Estructurada.....	58
4.2.2	Programación Orientada a Objetos:.....	59
4.3	Modelo de Ciclo de Vida y Paradigma de Programación Seleccionado....	61
5	<i>Análisis de Requerimientos.....</i>	64
5.1	Listado de Análisis Biogeográficos a Realizar.....	66
5.1.1	Diversidad.....	66
5.1.2	Estimadores de Riqueza (S_{est}).....	69
5.1.3	Índices de Similitud.....	71
5.1.4	Calidad de Muestreo (Q_s).....	75
5.1.5	Ligamientos.....	75
5.1.6	Elección de Áreas para la Conservación:.....	76
5.1.7	Localidades de Recolecta.....	78
5.1.8	Especies → Áreas.....	79

5.1.9	Áreas → Especies.....	79
5.1.10	UG's.....	80
5.1.11	Especies.....	81
5.1.12	Entre UG's.....	81
5.2	Comentarios del Listado de Análisis.....	82
5.3	Matriz de Decisión.....	83
5.4	Requerimientos de la Interfaz.....	86
5.4.1	Mapa :.....	87
5.4.2	Matriz coloreada:.....	88
5.4.3	Fenograma:.....	88
5.4.4	Pantallas de Salida:.....	88
5.4.5	Menús:.....	89
6	<i>Diseño</i>.....	91
6.1	Diseño de la Base de Datos.....	93
6.1.1	Los Datos de Recolecta:.....	93
6.1.2	Área del Geoframe:.....	94
6.1.3	Caminos Metodológicos:.....	95
6.1.4	Tablas de Rangos y Colores :.....	95
6.1.5	Tablas del Sistema:.....	96
6.2	Diseño de la Interfaz.....	96
6.3	Modelo de Objetos.....	98
6.3.1	La Arquitectura Final.....	101
6.4	Plan de Implementación.....	108
7	<i>Implementación</i>.....	111
7.1	Plataforma de Desarrollo.....	111
7.1.1	El Lenguaje de Desarrollo:.....	111
7.1.2	Open GL.....	114
7.1.3	La Base de Datos:.....	117
7.2	El Desarrollo.....	119
7.3	La Documentación.....	120
7.4	Las Siguietes Etapas.....	121

8	Conclusiones.	123
9	Apéndice 1 : Notaciones de Generales.	125
9.1	Notación de las Bases de Datos:.....	125
9.2	Notación de los Controles y Formas:.....	125
9.3	Notación de Programación y Lineamientos de Documentación.	126
10	Apéndice 2 : Diagrama de navegación	128
10.1	Archivo:.....	128
10.2	Edición.....	130
10.3	Ver.....	130
10.4	Análisis.....	130
10.5	Herramientas.....	133
11	Apéndice 3 : Base de Datos (Diagrama ER y Diccionario de Datos.)	136
11.1	Diagrama ER	136
11.2	Diccionario de Datos:	137
12	Apéndice 4 : Ejemplo de Documentación - Clase UG-	145
12.1	Descripción General:.....	145
12.2	Propiedades:.....	145
12.3	Funciones y Métodos:.....	145
13	Glosario	146
	Bibliografía:	148

1 Introducción

1.1 Objetivo:

Análisis y diseño de un Sistema de Información Geográfica (SIG) de aplicación específica, orientado a la Biología de la Conservación a través de los métodos aportados por la Biogeografía cuantitativa. El SIG ha de diseñarse de tal manera que sea una herramienta útil, poderosa y de manejo fácil para los usuarios.

1.2 Justificación :

Durante algunos cientos de miles de años el hombre ha tenido un papel ecológico central en la Tierra. Este papel no es compartido por ninguna otra especie, en el sentido de que nunca antes un grupo taxonómico había incidido tanto en la extinción de otros grupos. Recientemente se ha tomado conciencia de este problema, haciendo que ciertos grupos entiendan y se preocupen por nuestro entorno ecológico. La importancia de este conocimiento ha llevado a desarrollar una rama de la biología dedicada exclusivamente a la conservación y manejo de especies. Dentro de ella existe una variedad de disciplinas que aportan conocimientos distintos hacia una estrategia de conservación. Todas las ramas de la ecología, como la genética, la biosistemática y la biogeografía, tienen conocimientos que deben ser utilizados por un estudioso de la conservación (Eguiarte y Piñero ,1990).

La biogeografía es la parte de la biología que estudia la distribución geográfica de las especies. Una de las clasificaciones más usadas para la biogeografía es aquella que la divide en "histórica" y "ecológica". La primera estudia la distribución de la biota a escalas geológicas. La segunda considera procesos en una escala menor de tiempo y explica la distribución de especies mediante factores ecológicos y no

geológicos. La biogeografía ecológica nos permite saber qué áreas son importantes, en función a la cantidad y calidad de especies que la habitan.

Las tecnologías de la información pueden ser herramientas muy poderosas en análisis biogeográficos, permitiendo al investigador dedicar más tiempo al estudio y análisis de los resultados que a la generación de los mismos, ya que las computadoras se han distinguido, desde la década de los 60's, como poderosas herramientas para el manejo de grandes volúmenes de información, gran velocidad, además de que su manejo se ha vuelto cada vez más sencillo para la mayoría de las personas.

La computadora es hoy en día lo que fue la máquina de vapor para la revolución industrial: La máquina de vapor permitió potenciar la fuerza humana, la computadora está potenciando nuestro intelecto.

Una de las áreas de los sistemas de información que se ha desarrollado recientemente, es la referente a los Sistemas de Información Geográfica (SIG). Estos sistemas han revolucionado el análisis de datos de contexto espacial, al aplicar todas las ventajas que ofrece un sistema informático en la generación y análisis de mapas, y es precisamente esta área en la que se pretende proporcionar una herramienta a la biología de la conservación para la realización de análisis biogeográficos y emprender acciones urgentes que permitan hacer compatibles las necesidades crecientes de las poblaciones humanas con la necesidad de conservar los hábitats más amenazados, y de usar en forma adecuada y sostenida los hábitats explotables. De otra manera, las futuras generaciones jamás entenderán cómo pudimos heredarles un patrimonio cultural y tecnológico tan vasto y un patrimonio natural tan degradado (Escurrea, 1990).

1.3 Planteamiento del Problema:

Aplicar la metodología de desarrollo de sistemas más apropiada, para generar un Sistema de Información Geográfica (SIG) que permita realizar los análisis

biogeográficos más importantes actualmente, permitiendo automatizar los procesos de captura y cálculos necesarios para la obtención de los resultados pertinentes.

El alcance de la presente tesis es generar el análisis y diseño de un software comercial, el cual será una herramienta para análisis biogeográfico mediante los métodos de la biogeografía cuantitativa que permite realizar con facilidad exploraciones empíricas, despliegues gráficos de la información y análisis.

Este software permitirá realizar análisis biogeográficos con únicamente la información de recolección de la especie: nombre de la especie, latitud y longitud de la recolecta. Puede o no utilizarse cartografía digital.

Con la herramienta creada se podrán calcular varios parámetros ecológicos relevantes como la diversidad alfa, beta y gama, estimación de la riqueza de una determinada área de estudio (AE.), etc. Se podrán obtener mapas de distribución de un conjunto de especies, así como mapas de riqueza. También se podrán clasificar y jerarquizar subregiones biogeográficas al AE mediante análisis y agrupamientos, teniendo así elementos cualitativos y cuantitativos para proteger un área determinada, lo cual permita conservar la máxima riqueza biológica en beneficio de todos los que habitamos este planeta.

2 Sistemas de Información Geográfica

2.1 ¿Qué es un Sistema de Información Geográfica (SIG)?

Un Sistema de Información Geográfica (SIG o GIS por Geographical Information System) es un sistema que utiliza recursos de cómputo (Software y Hardware) para efectuar tareas de captura, almacenamiento, manipulación, modelado, análisis y visualización de información referenciada geográficamente con el objeto de resolver problemas complejos en aspectos geográficos. Los SIG's han revolucionado la manera en que la información geográfica (como son los mapas, fotografías aéreas, imágenes de satélite, estadísticas geográficas, etc.) es usada en la toma de decisiones, investigación, administración, planificación, administración e investigación.

En la última década la funcionalidad que nos ofrece un SIG ha propiciado que cada día mayor número de organizaciones aprovechen las ventajas de representar espacialmente la información. Para muchas de estas organizaciones la ubicación se ha convertido en un concepto crítico para la utilización óptima de todos sus recursos (dinero, tiempo, personal, instalaciones, etc.).

La idea de representar la información en un contexto espacial no es nueva, pero la utilización de recursos de cómputo (los cuales son más potentes y baratos cada día) han revolucionado el manejo de esta información de la siguiente manera:

1. *El Almacenamiento:* La información es guardada en medios magnéticos de manera digital, que a diferencia de los métodos tradicionales tales como mapas y archivos impresos en papel, ocupan mucho menor espacio físico, son fácilmente duplicables y más resistentes al paso del tiempo pues conservan de mejor manera la información.

2. *La Consulta de la información:* Se tiene a la mano en tiempos muy cortos, permitiendo diferentes grados de acercamiento, además de poder representar una gran variedad de información.
3. *Captura y modificación de información:* Permite captura de información geográfica a través de diversos medios (fotos, satélites, GPS, etc.), permitiendo además modificar dicha información sin tener que redibujar el plano en papel.
4. *El Análisis de Información:* Utilizando el poder de procesamiento de grandes cantidades de información que poseen los sistemas de cómputo, se genera información nueva a partir de un conjunto de datos.

Estas ventajas son las que presentan cualquier Sistema de Información (SI), pero lo que hace diferente a un SIG son sus estructuras de datos orientadas a la localización geográfica que dan la capacidad de combinar, analizar y mostrar información diversa permitiendo agregar dimensiones extras para una mejor comprensión.

La necesidad de datos y análisis espaciales no están restringidas únicamente a los investigadores de las Ciencias de la Tierra. Diversas instituciones involucradas en la planeación urbana y el catastro necesitan información detallada sobre la distribución de tierra, sus recursos y usos; los ingenieros civiles necesitan un plan de rutas de las carreteras y canales para estimar costos; los departamentos de policía necesitan mapas de distribución de delitos; las organizaciones médicas necesitan las distribuciones de enfermedades; las empresas de lucro necesitan saber la distribución de sus sucursales y la capacidad del mercado. La enorme infraestructura que conocemos generalmente como servicios -luz, agua, telefonía- puede ser almacenada y manipulada en forma de mapa.

2.2 Conceptos de SIG.

Antes de abordar el tema de la presente tesis, sería conveniente explicar algunos conceptos básicos, en los cuales se basa y/o utiliza un SIG. No es necesario ser un experto en geografía pues sólo se dará un bosquejo de éstos, lo que nos permitirá una mejor comprensión.

2.2.1 Geodesia.

Geodesia (*γηος = tierra, δαιω = divido*) es la ciencia de medir y cartografiar la superficie terrestre (Helmert A., 1880). Esta definición ha conservado su validez hasta ahora; incluye la determinación del campo de gravedad externo de la Tierra, así como la superficie del fondo del mar. Con esta definición, la geodesia se puede incluir en las geociencias y también en las ciencias de la ingeniería.

La geodesia se puede dividir en las áreas de geodesia global, levantamientos geodésicos y topografía. La geodesia global es la responsable de determinar la figura de la Tierra, incluyendo el campo de gravedad completo. Un levantamiento geodésico define la superficie de un país mediante coordenadas de un número suficientemente grande de puntos de control. En este trabajo fundamental se debe de considerar la curvatura global de la Tierra. En la topografía (levantamientos topográficos, catastrales y de ingeniería), se obtienen detalles de la superficie del terreno; en general el plano horizontal es suficiente como plano de referencia.

2.2.2 Cartografía.

Según la definición adoptada por la comisión para la formación de cartógrafos de la Asociación Cartográfica Internacional, reunida por la UNESCO en París, en abril de 1966, la cartografía:

“Tiene por objeto la concepción, preparación, redacción y realización de los mapas. Comprende el conjunto de estudios y operaciones científicas, artísticas y técnicas que intervienen a partir de los resultados de las observaciones

directas o de la explotación de una documentación, en el establecimiento de mapas, planos y otras formas de expresión, así como en su utilización" (Joly, 1979).

En la realización de un mapa están involucradas varias operaciones necesarias, desde las mediciones directas sobre el terreno (lo que se llama levantamiento), la recolección de información escrita que se traducirá en un mapa, hasta la impresión definitiva y la difusión del documento cartográfico. Es así que la cartografía es considerada una ciencia, un arte y una técnica.

A través de los siglos el hombre ha ido perfeccionando su conocimiento de la Tierra y ha podido conceptualizar varias nociones que son el fundamento de la cartografía moderna, como son:

- La esfericidad de la Tierra.
- La determinación astronómica de las latitudes y las longitudes.
- El sistema de coordenadas terrestres.
- Los sistemas de proyección.

Gracias a estas nociones es posible realizar una representación geométrica plana, simplificada y convencional, de toda o parte de la superficie terrestre, con una relación de similitud proporcionada a la que se llama escala, es decir, un mapa.

2.2.3 Cartografía por Computadora.

La cartografía por computadora es la primera aportación de las Tecnologías de la Información (TI) en beneficio de las geociencias, pues dichas TI automatizan algunas partes de los procedimientos existentes para la creación y presentación de un mapa. Pero se ha visto que las TI no sólo sirven para automatizar procesos ya existentes, sino que son capaces de crear nuevos, o bien efectuar algunos que ya se conocían pero se carecía de los recursos para hacerlo, siendo posible también que las

TI analicen información pudiendo inclusive predecir patrones. Las razones para utilizar la computadora en la creación de mapas son publicadas por Rhind en 1977:

1. Para hacer los mapas existentes más rápido.
2. Para hacer los mapas existentes más baratos.
3. Para hacer mapas para necesidades específicas.
4. Para hacer posible una producción de mapas, cuando se carece de los recursos humanos para hacerlo.
5. Para permitir la experimentación con diferentes representaciones de los mismos datos.
6. Para facilitar la creación y actualización de mapas, cuando los datos ya se encuentran en forma digital.
7. Para facilitar análisis de datos que demandan interacción entre análisis estadísticos y mapas.
8. Para minimizar el uso de mapas impresos para almacenar información y minimizar además los efectos de la clasificación y generalización en la calidad de la información.
9. Para la creación de mapas que son difíciles de efectuar a mano (ej. Mapas en 3D).
10. Para crear mapas en donde los procedimientos de selección y generalización están explícita y consistentemente aplicados.
11. La introducción de la automatización permite dirigir una revisión a todo el proceso de creación de mapas, lo que puede generar ahorros y mejoras en dichos procesos.

La cartografía por computadora es la automatización de un procedimiento mecánico, pero no hace nada nuevo más que la creación de mapas, aunque la adición de una simple capacidad de superposición, permite a los usuarios iniciar con poco esfuerzo una serie de análisis lógicos y espaciales en información mapeada, como estudios de planeación (ej. Steinnitz y Browb 1981) o ecológicos (Luder 1980) que anteriormente eran muy difíciles de hacer a mano. Cuando la cartografía por

computadora es combinada con herramientas de administración, como en los sistemas de diseño asistido por computadora (CAD por Computer Aided Design) donde estas herramientas parten de la imagen para generar la lista de partes necesaria, la cartografía por computadora es así descrita como Cartografía Automatizada/Gestión de instalaciones (AM/FM por Automated Mapping/Facility Management) lo cual es un enfoque visual de un sistema de inventario.

Es en este punto donde la cartografía por computadora da origen a los SIG, pues ella solo se queda en la presentación de la información en mapas mediante una pantalla y el SIG va un paso más adelante al poder integrar y analizar información no solamente para la creación del plano, sino información que tiene algún componente de localización.

Esto ha propiciado que las metodologías que combinaban y analizaban información en un marco geográfico, sean implementadas en algoritmos y modelos para computadora dando un desarrollo cíclico como el siguiente (Figura 2.1):

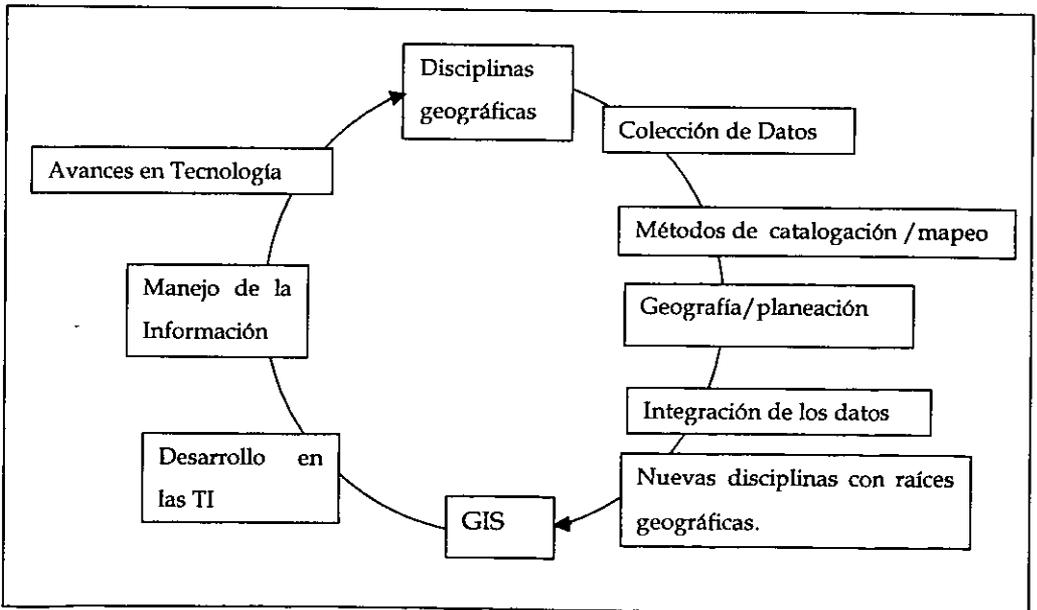


Figura 2.1 Desarrollo cíclico de un SIG.

2.2.4 Mapas.

Un mapa es un modelo generalizado, plano y a escala de información del espacio. Esta información es cualquier tipo de dato que involucre un aspecto de localización o extensión. Un modelo es una abstracción acerca de algún aspecto de la realidad, donde típicamente se dejan fuera detalles con el objeto de simplificar.

Un mapa representa características geográficas o bien fenómenos espaciales mediante la graficación adecuada de información referente a sitios y atributos. También se le define como un conjunto de puntos, líneas y áreas que están definidas por la localización en el espacio a través de un sistema de referencia coordinado y por sus atributos no espaciales.

Un mapa es generalmente representado en 2 dimensiones, pero no hay razón de excluir mayores dimensiones excepto por la dificultad de llevarlas a una pieza de papel.

2.2.4.1 Tipos de Mapas

Existen varios tipos de mapas, pero los de nuestro interés son los siguientes:

- Mapas Topográficos

Son la representación más exacta y detallada de la superficie terrestre, en cuanto a la posición, forma, dimensiones e identificación de los accidentes del terreno, así como de los objetos concretos que se encuentran permanentemente sobre él. Los mapas topográficos son generalmente mapas base y de propósito general.

- Mapas Temáticos

Representación convencional de distintos fenómenos localizables y sus correlaciones que se expresan mediante símbolos cualitativos y cuantitativos, sobre un fondo de referencia, es decir un mapa topográfico.

Un mapa temático no modela la superficie de la tierra, pero presenta información temática de tal forma que su relación con la superficie terrestre es reflejada correctamente. Los temas que se representan pueden ser fenómenos humanos (ej. Nivel socioeconómico) o naturales (ej. temperatura del suelo).

La producción de este tipo de mapas ha generado mucho interés debido a que nos ofrece grandes posibilidades en el análisis de datos.

2.2.4.2 *Simbología.*

Los mapas plasman la información a través de una determinada simbología. Los símbolos son los elementos que se dibujan sobre el mapa, además del relieve para representar los diversos objetos o temas que se encuentren localizados dentro del mapa y pueden ser:

- Puntuales: son la representación discreta de la localización de un objeto.
- Lineales: Son símbolos que nos permiten marcar redes, fronteras políticas y económicas.
- De área: Nos permiten delimitar extensiones de terreno.
- Texto : Nombres y números que identifican lo que está dibujado en el mapa.
- Colores: El color ayuda en gran medida en la diferenciación de objetos similares dentro del plano.

2.2.4.3 *Escala.*

Como ya se mencionó, un mapa es un modelo a escala; la escala se refiere al poder representar al mundo real en un espacio menor. En el caso de los mapas la escala nos proporciona un factor por el cual deben ser multiplicadas las distancias en el mapa para obtener la distancia real en la superficie terrestre. Lo anterior no es tan válido cuando la superficie modelada es muy grande debido a la forma de nuestro planeta (tema que tocaremos en las proyecciones).

En SIG los mapas son almacenados de tal forma que podremos variar la escala de manera dinámica, es decir, podemos manipular el dispositivo de salida para poder alterar la escala con la que estamos viendo la información sin alterar ningún dato, por lo que el concepto de escala solamente está presente cuando el mapa es visto a través de un dispositivo de salida (en pantalla y/o impreso).

2.2.4.4 Generalización.

Pero esto no quiere decir que al disminuir la escala (inclusive eliminarla 1:1), en un GIS veamos mayor detalle dentro del mapa, pues este adolece de los inconvenientes de ser un modelo; la información con la que se crea un mapa es abstraída, es decir, se dejan fuera ciertos detalles, esto en cartografía se denomina generalización.

La generalización puede unir elementos y formar uno solo a lo que se le denomina combinación; puede eliminar detalles lo que es la simplificación; puede representar con símbolos elementos similares con atributos diferentes a lo que se le denomina clasificación e inclusive puede no representarlos por lo que los selecciona.

Otro aspecto importante en los mapas es que algunos de sus elementos deben ser exagerados, es decir, son representados con mayor tamaño al que serían representados a la escala correspondiente, y esto inclusive propicia que algunos elementos sean movidos para ser vistos claramente, a lo que se le denomina desplazamiento.

Esta generalización en los mapas impresos proviene de la escala, y de la cantidad y calidad de la información que se desea representar. En un mapa impreso con una escala de 1:100, 000 no se debe esperar una precisión de localización de más de 1 mm. Como la fuente de información cartográfica de un GIS son generalmente mapas impresos que son digitalizados, es absurdo pensar que éstos van a ser más precisos que los impresos.

2.2.4.5 *Proyecciones.*

Si la Tierra tuviese forma cilíndrica o cónica, la cartografía no comprendería tan variado conjunto de problemas de incompleta y no satisfactoria solución. Tanto la superficie cilíndrica como la cónica pertenecen a cuerpos redondos desarrollables¹, mientras la esfera y los esferoides no gozan de esta propiedad. De aquí se deduce la imposibilidad de representar exactamente su superficie en un plano y habrá necesidad de utilizar un artificio u otro que nos permita formarnos una idea aproximada de la totalidad de la superficie o de la extensión que deseemos estudiar.

En realidad, el mejor procedimiento productivo es el de los globos terráqueos, en los cuales, hecha previamente la reducción de escala, podemos representar con toda exactitud los accidentes geográficos: sin embargo, este procedimiento no tiene más aplicación que la propia de las tareas pedagógicas elementales, toda vez que sería imposible construir un globo terráqueo de las dimensiones adecuadas.

Antes de explicar las proyecciones es necesario comprender que la localización de un punto sobre nuestro planeta, se hace con coordenadas esféricas (2 ángulos y un radio), porque son la mejor opción debido a la forma de la tierra. Además si hacemos la simplificación de que la tierra es esférica² y por tanto el radio siempre es el mismo y solamente tomamos una medida de elevación con respecto al nivel medio del mar (lo que viene siendo la altitud).

Los 2 ángulos en cartografía para la localización de un punto son: latitud y longitud. La latitud es el ángulo medido en el plano ecuatorial del elipsoide y la

¹ Un cuerpo desarrollable puede explicarse de manera sencilla como aquel que puede ser construido con una hoja de papel (Un plano el cual se deforma), el cono y el cilindro son cuerpos desarrollables no así la esfera.

² La Forma de la Tierra es parte del estudio de la Geodesia. Dado que la Tierra no es una esfera, pues se encuentra achatada de los polos, se le ha dado a la forma de nuestro planeta el nombre propio de Geoide, el cual no puede ser descrito con una fórmula matemática, pero se puede aproximar con los denominados elipsoides como por ejemplo el de Hayford el cual no difiere más de 50 m del Geoide.

superficie normal en el punto a localizar. La longitud es el ángulo medido entre el meridiano cero y el plano del meridiano del punto a localizar.

Los puntos que tienen la misma latitud forman un plano que corta a la Tierra en círculos horizontales, de los cuales el mayor es el ecuador, y se van reduciendo hacia los polos y son llamados paralelos. En cambio los puntos que tienen la misma Longitud, son los denominados meridianos, todos son del mismo radio y se selecciona uno de manera arbitraria para fungir como el meridiano cero (el de Greenwich).

Las proyecciones son un conjunto de principios y reglas matemáticas que nos permiten representar un punto ubicado sobre la superficie de nuestro planeta, en un plano. Estos modelos matemáticos distorsionan al menos una de las propiedades como la forma, el área, la distancia y/o dirección.

Algunas proyecciones son:

- Cónicas: Se distinguen a primera vista porque en ellas los paralelos están representados por arcos de circunferencias concéntricas. En el centro común de estas circunferencias concurren una serie de rectas, los meridianos, que están representados por una serie de generatrices del cono de proyección.
- Cilíndricas: Pueden considerarse como un caso límite de las cónicas en que el cono de proyección fuese tangente al ecuador, convirtiéndose en cilindro. Se distingue en todas las demás en que los meridianos -suponiendo que la proyección sea polar-, se dibujan paralelamente y a la misma distancia uno de otro. Esto trae como consecuencia una gran deformación superficial a medida que aumenta la latitud. La proyección de Mercator (muy utilizada en la navegación) es una proyección cilíndrica, pero los paralelos no se encuentran equidistantes sino que la separación entre dos consecutivos aumenta en razón inversa al coseno de la latitud.
- UTM (Universal Transverse Mercator) Para el sistema UTM, el globo está dividido en 60 zonas, cada zona se extiende a 6 grados de longitud. Cada zona

tiene su propio meridiano central. No se considera una proyección sino un sistema de coordenadas.

2.2.5 Tipos de SIG.

Los SIG dependiendo de la funcionalidad y flexibilidad que presenten pueden clasificarse en:

2.2.5.1 *De aplicación Genérica.*

En estos sistemas se tienen todas las funciones de un SIG: captura, almacenamiento, manipulación, modelado, análisis y visualización, ofreciendo una estructura básica la cual puede ser ampliada para que se adapte a nuestras necesidades. Pero tienen el inconveniente de ser más difíciles de manejar, de demandar muchos recursos y generalmente ofrecen lenguajes interpretados (macros) para el desarrollo de aplicaciones específicas.

2.2.5.2 *De aplicación Específica.*

En estos sistemas algunas de las funciones propias de un SIG, no se implementan en función de ofrecer un manejo más sencillo para el usuario, reducir la complejidad, acotar el problema, etc. Generalmente son programas compilados en su totalidad, por lo que su ejecución es más veloz que las macros. Por ejemplo un SIG específico podría ser un sistema de transporte urbano, el cual permitiera capturar y ver las rutas de los diferentes medios de transporte en una ciudad (en la Cd. de México: Metro, rutas de microbuses, rutas de camiones, trolebús y tren ligero) sobre una cartografía base; podría además analizar esta información para dar la mejor ruta en función del tiempo empleado. En un SIG genérico podríamos hacer un análisis de la mejor ruta en función del tiempo agregando la estructura y los datos pertinentes, además de modificar el algoritmo de búsqueda para contemplar este nuevo parámetro.

Estas adiciones en un SIG específico no se pueden hacer sin rediseñar el programa y recompilarlo, y al igual que el SIG genérico estas adiciones también son complejas de implementar.

Hay otra clasificación que corresponde a la manera en que manejan y representan la información gráfica, y son los denominados SIG vectoriales y Raster, los cuales se explicarán más adelante en este mismo capítulo.

2.2.6 Antecedentes de SIG.

Desde las civilizaciones más antiguas hasta la actualidad, los datos espaciales han sido recolectados por navegantes, geógrafos, viajeros y supervisores que han plasmado dicha información en forma gráfica por medio de la creación de mapas.

El relacionar información con un aspecto de localización en un mapa, no es nada nuevo, se conocen pinturas rupestres de hace 5,000 años antes de Cristo con mapas que tienen además registros de caza.

El libro de Domesday Book es un detallado inventario de Inglaterra, hecho en 1806 por la orden de William the Conqueror. Dicho libro provee información de la población, el uso de la tierra y sus dueños con la finalidad del cálculo de impuestos y exploración.

En la década de 1960, el gobierno de Canadá a través de su Agencia de Energía, desarrolla el primer SIG, en el cual se unieron los datos recopilados por el sistema de inventario de tierras con las tecnologías de cartografía automatizadas para dar origen al primer SIG en 1962. El sistema fue diseñado para identificar tierras con potencial de ser explotadas en la agricultura y actividades forestales utilizando mapas que contenían información sobre el clima y formaciones geográficas. Además, las tierras eran clasificadas bajo criterios de ciertas especies de la fauna en asociación a ciertas características ecológicas como áreas propensas a ser utilizadas como de conservación y recreativas.

A lo largo de las décadas de los 60's y 70's los SIG fueron construidos para situaciones estrictamente ligadas a necesidades de alguna institución pública. Se trataba de sistemas instalados en grandes computadoras, cuyo objetivo era el inventario de recursos, con poca capacidad de análisis. Pero en la década de los 80's y especialmente en la de los 90's los SIG se han convertido en un tema de gran relevancia que con el rápido desarrollo del Hardware, al hacerse cada vez más poderoso y barato, ha propiciado un desarrollo más acelerado permitiendo además que este tipo de tecnología esté cada vez más a la mano de diversas instituciones.



Figura 2.2 Integración de los 3 componentes de un SIG.

2.2.7 Estructura de un SIG.

Un SIG visto desde una perspectiva técnica, es un sistema que integra fuertemente una base de datos con información espacial, un motor gráfico para el despliegue de los mapas, pudiendo ser un CAD, y herramientas de análisis de la información (Figura2.2).

No basta tener los 3 componentes para decir que poseemos un SIG, y hacemos énfasis en la integración de ellos, la base de datos debe proporcionar la información que ha de ser analizada por las herramientas de análisis, para su posterior visualización referenciada geográficamente.

2.2.7.1 *Motor gráfico.*

El motor gráfico en un SIG es la parte encargada de desplegar la información contenida en la Base de Datos y/o en los archivos, en un dispositivo de la computadora, ya sea en la pantalla, en la impresora o en algún otro medio. El motor gráfico de un SIG es el conjunto de instrucciones que convierte los datos en un mapa, captura las peticiones del usuario (Zoom, Pan, etc.) y es una parte de la captura de datos y mapas.

Generalmente en esta parte se utiliza y/o se toman muchos aspectos de la funcionalidad que ofrece un CAD para la implementación del motor gráfico, siempre pensando en su integración con los demás elementos que conforman el SIG.

Los sistemas de información geográfica tienen muchas cosas en común con los sistemas de CAD utilizados en el diseño una gran variedad de objetos técnicos, desde aviones hasta microchips. Ambos, GIS y CAD, necesitan ubicar los objetos en una referencia, manejar datos no gráficos y describir relaciones topológicas. La mayor diferencia que tiene un GIS con un CAD es el volumen y gran diversidad de datos que maneja el GIS y de la naturaleza de sus análisis especializados. Las diferencias pueden ser tan grandes que hacen que un eficiente CAD sea totalmente inadecuado para un GIS y viceversa.

Hay 2 maneras de manejar gráficos por computadora, y las dos son capaces de generar los mismos resultados (claro que a diferentes costos), es decir, si una persona observa una imagen en un monitor no podría asegurar como fue generada y como

está guardada, a menos que haga ciertas operaciones con ella o intuya por su complejidad la manera en que se generó.

Estas dos maneras de generar gráficos son:

1. Imagen Raster.
2. Dibujo Vectorial.

2.2.7.1.1 Dibujo vectorial vs. Imagen Raster.

Vamos a analizar como la computadora es capaz de generar gráficos, explicaremos cada uno de ellos y analizaremos sus pros y contras y para qué tipos de gráficos son adecuados:

Las imágenes raster son en realidad una matriz de pequeñas áreas que conforman la imagen, cada celda de la matriz alberga atributos de esta área por ejemplo el color, el brillo, etc. La selección del tamaño de esta área tiene que ver con los siguientes aspectos:

1. Las limitaciones físicas del dispositivo: Esto se refiere al área de menor tamaño que puede distinguir, y se le denomina resolución, midiéndose en puntos por pulgada (PP). Podemos tener una imagen con una resolución mayor a la del dispositivo, pero al momento de desplegarla en dicho dispositivo no podremos ver esa resolución extra.
2. La memoria que ocupa: entre más pequeña sea dicha área, mayor memoria requerirá para su almacenamiento y despliegue pero mayor será el nivel de detalle que será capaz de mostrar. Actualmente existen algoritmos que optimizan esta área a través de la eliminación de la redundancia de información en la matriz.

Otro factor importante en el cálculo del tamaño de una imagen raster, es la profundidad de color. Cuanto mayor sea la codificación del color, se puede disponer de una mayor cantidad de colores para cada área (2 colores se pueden representar con 1 bit, 4 con 2) pero la cantidad de memoria para almacenar la imagen es también mucho mayor. Este aspecto es sujeto también a optimización, a través de diferentes métodos (tablas de colores, generación de un color por variación en la densidad de los colores existentes, etc.).

Una manera de calcular el espacio de memoria que ocupa una imagen en KB, sin tener ninguna compresión es:

$$KB = \frac{RV * RH * Nbc}{1024 * 8}$$

Donde : KB: es la memoria en Kilobytes.

RV: es el número de renglones de la matriz.

RH: el número de columnas de la matriz.

Nbc: Es el número de bits utilizados para la codificación del color.

*1024*8*: Es el número de bits en un KB.

En realidad, una imagen raster es la manera final en la que la computadora muestra la información gráfica en el monitor, pues la pantalla es una matriz de pequeñas áreas, denominadas pixeles (Proviene de la unión de 2 palabras inglesas: Picture Elements), el cual le corresponde a una determinada área de memoria dentro de la memoria RAM de la computadora para almacenar su información, y la cual es leída por el sistema de vídeo para su despliegue.

El formato raster es muy útil para imágenes muy complejas, como podría ser una fotografía, pero tiene el inconveniente de que si se les cambia la escala (se hace un acercamiento) llega un punto en el cual se es capaz de distinguir los pixeles perdiéndose por ello detalle de la imagen.

Para los SIG esta manera de generar imágenes está muy relacionada con la estructura de su base de datos, y el tipo de elementos que puede almacenar. La manera más simple consiste en un arreglo de celdas en una rejilla. El tamaño de la celda está en relación directa con la capacidad de almacenamiento y despliegue, y una mala selección de su tamaño puede traer consecuencias en el agrupamiento de la información; sin embargo, esta forma de manejar la información, permite algoritmos sencillos y rápidos en cálculos y operaciones lógicas con la superposición de áreas (unión, intersección y negación).

La otra manera de generar gráficos en la computadora, es mediante la vectorización: esta forma en vez de almacenar toda la matriz, lo que guarda son los comandos para dibujar dentro de esta matriz, por ejemplo:

Para dibujar una línea la imagen raster contendrá toda la matriz de pixeles, con todos los pixeles apagados a excepción de aquellos que estén dentro de la línea, en cambio el dibujo vectorial contendrá el comando para trazar la línea, que deberá ser interpretado por el CPU, y el punto inicial y final de ésta.

Los limitantes para este tipo de generación de imágenes tienen que ver con la complejidad del dibujo, por ejemplo una fotografía no sería apropiado vectorizarla, debido a que para obtener el mismo grado de detalle que una imagen raster, serían necesarios demasiados comandos, sobre todo con el manejo de los colores.

Pero una ventaja de este tipo de generación de imágenes es que prácticamente no hay límite para el nivel de acercamiento, la información para la generación de la gráfica puede ser almacenada dentro de una base de datos y se pueden implementar de manera más sencilla análisis matemáticos a los elementos gráficos, como por ejemplo el análisis de redes.

Algunos de los inconvenientes de esta forma de representar la información, es en la definición de polígonos, donde podríamos tener trayectos muertos, duplicidad de información en polígonos contiguos y algoritmos complejos para el cálculo de y superposición de áreas.

A continuación se muestran dos tablas con las ventajas y desventajas de las dos maneras en que puede trabajar un SIG:

RASTER:

Ventajas	Desventajas
<ul style="list-style-type: none"> • Estructura simple. 	<ul style="list-style-type: none"> • Los volúmenes de datos que se manejan.
<ul style="list-style-type: none"> • La superposición y combinación de datos mapeados con información censada remotamente es fácil. 	<ul style="list-style-type: none"> • El uso de celdas grandes para la reducción del volumen de la información puede traer serias pérdidas de información, y por ende una mala comprensión del fenómeno a medir.
<ul style="list-style-type: none"> • Diferentes tipos de análisis espaciales pueden ser implementados. 	<ul style="list-style-type: none"> • Los mapas raster son menos estéticos que los mapas vectoriales.
<ul style="list-style-type: none"> • Las simulaciones son fáciles debido a que las unidades tienen la misma forma y tamaño. 	<ul style="list-style-type: none"> • Las redes son difíciles de implementar.
<ul style="list-style-type: none"> • La tecnología se convierte cada vez más barata y poderosa. 	<ul style="list-style-type: none"> • Los cambios de proyección son muy demandantes de recursos y tiempo.

VECTOR:

Ventajas	Desventajas
<ul style="list-style-type: none"> • Buena representación del fenómeno. 	<ul style="list-style-type: none"> • Complejas estructuras de datos.
<ul style="list-style-type: none"> • Estructura compacta. 	<ul style="list-style-type: none"> • Dificultad en la combinación de varios mapas de polígonos o vectores y rasters en superposiciones.
<ul style="list-style-type: none"> • Las topologías son fáciles de implementar con redes. 	<ul style="list-style-type: none"> • La simulación es difícil debido a que cada unidad es diferente.
<ul style="list-style-type: none"> • Gráficas precisas. 	<ul style="list-style-type: none"> • El despliegue y el ploteo es caro, especialmente para una alta calidad.
<ul style="list-style-type: none"> • Es posible la recuperación, actualización y generalización de la información. 	<ul style="list-style-type: none"> • La tecnología para este tipo de gráficas es muy especializada.
	<ul style="list-style-type: none"> • Análisis espaciales y filtrado son imposibles.

En años recientes se ha visto que lo que se consideraba un problema conceptual (raster-vector), es en realidad un problema tecnológico. Para los GIS de tipo raster, su limitante es la capacidad de almacenamiento, pues los algoritmos de

cálculo de áreas, sumas, promedios, y otras operaciones son inclusive más eficientes que en una estructura de vectores, pues se reducen a simples conteos. En cambio modelar una red en una estructura de vector es relativamente fácil, pero también su tamaño se incrementa fuertemente cuando la red es más compleja y contiene mucha información sobre la topología.

Si en un mismo sistema podemos convertir una estructura a otra, según la necesidad que tengamos, el problema de raster o vector desaparece, por lo que el sistema en cuestión utiliza las dos estructuras para representar información espacial, y ambas estructuras son inter-convertibles. Las conversiones de vector a raster son realizadas por hardware a muy altas velocidades (se puede comprobar esto en cualquier monitor), en cambio la operación inversa no es de todo bien comprendida, y aunque hay una serie de adelantos muy fuertes todavía quedan algunos problemas a resolver (ej. dos líneas demasiado cerca una de otra). Por lo que debemos ver, las dos estructuras no como antagonistas sino como complementarias.

2.2.7.1.2 Concepto de Punto.

En los mapas, y debido a la generalización que tenemos que hacer, los puntos representan elementos geográficos, con áreas menores o mayores de las que el punto ocupa gráficamente dentro del mapa.

Para fines de un SIG, el punto es la manera de localizar un determinado elemento dentro de un mapa; en caso de imágenes vectoriales, el punto es el vértice de un polígono, es el lugar donde un trayecto cambia de dirección, el centro de un círculo, etc.

En las imágenes raster el punto es un área específica de la matriz, y si algo es menor al área de la celda de la matriz, y se necesita ver, se representará con una celda completa, de ahí el problema de la exactitud en la localización y del tamaño de un elemento gráfico en las imágenes raster. En cambio un punto dentro de una imagen vectorial es almacenado como un par ordenado de números (en 2D ó 3 en 3D), los

cuales pueden ser tan exactos como nosotros queramos³. En una BD, los puntos pueden ser almacenados junto con atributos diferentes a la localización, de la siguiente manera (Figura 2.3):

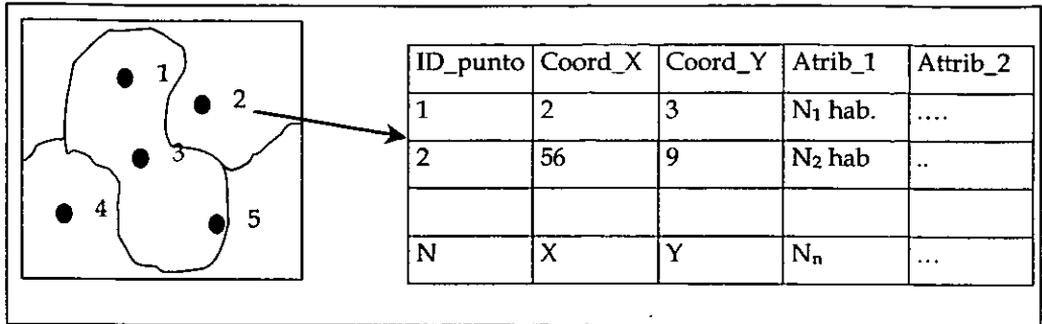


Figura 2.3 Almacenamiento de los puntos.

2.2.7.1.3 Concepto de Línea.

Las líneas en un SIG se definen más bien bajo el esquema de un segmento, el cual va de un punto a otro, pero no necesariamente en una línea recta, podría ser un segmento de arco o de cualquier otra curva.

En las imágenes raster una línea es una sucesión de celdas contiguas que tienen algún atributo común o la frontera entre 2 ó más áreas que tienen la misma particularidad. En cambio en las imágenes vectoriales la línea es el elemento gráfico que une dos puntos (forma arco-nodo), teniéndose como ventaja, con respecto a la línea raster, que podemos definir tipos de línea discontinuos y de diferentes grosores.

³ No necesariamente precisos, ya que la exactitud se refiere al grado de fineza con la que un objeto puede medirse, y la precisión a que esta medida sea real. Además que aquí cabe agregar que muchas veces, por el poder que nos ofrece la electrónica, manejamos una exactitud extrema en los cálculos, lo cual está bien, pero en los resultados esta exactitud carece de sentido si no podemos aplicarla en el mundo real.

En algunos motores gráficos de los SIG, al igual que algunos CAD, sobre todo los menos recientes, las curvas son representadas con pequeños segmentos de recta, teniendo consecuencias en los cálculos de áreas y en espacio de almacenamiento, por lo que los paquetes más recientes definen los segmentos curvos como tales, a través de ecuaciones matemáticas paramétricas, solamente ocupándose el esquema de pequeñas rectas en el intercambio de información entre diferentes formatos digitales.

De una manera similar, las líneas pueden ser guardadas en una BD, con atributos extras a la manera de dibujar la línea, para su consulta y/o despliegue (Figura 2.4).

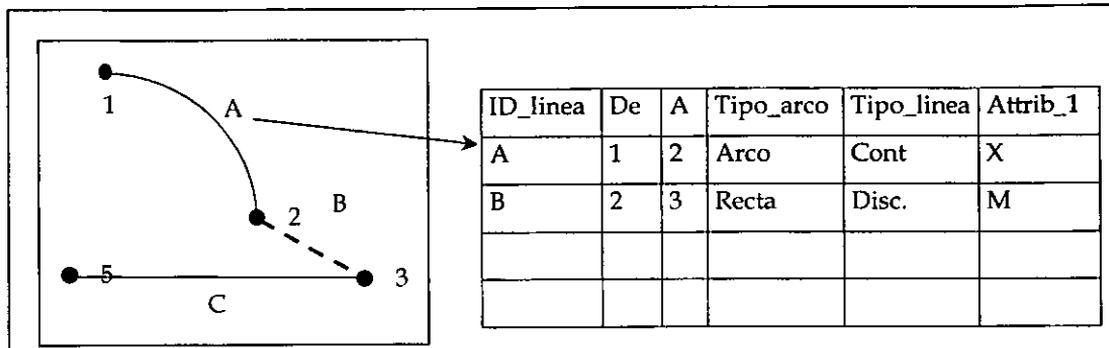


Figura 2.4 Estructura arco nodo.

2.2.7.1.4 Concepto de Polígono.

Un polígono es un conjunto de segmentos, los cuales delimitan una determinada área. En un mapa representa un área con una característica común, una frontera, el área de estudio, etc.

En una imagen raster los polígonos, y en general las superficies, son un conjunto de celdas contiguas entre sí, en cambio en un formato vectorial son la lista de puntos y segmentos que forman el perímetro de la figura.

Dentro de una BD los polígonos, pueden ser almacenados bajo un esquema similar al de la línea (Figura 2.5):

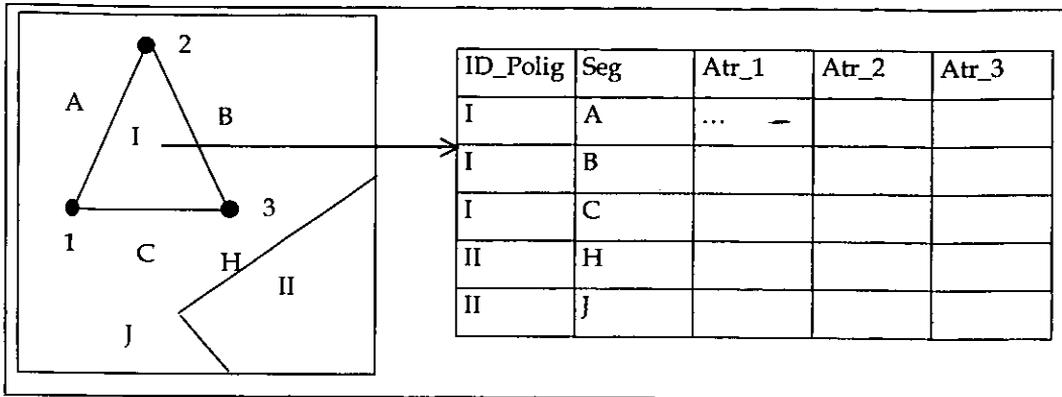


Figura 2.5 Almacenamiento vectorial de polígonos.

2.2.7.1.5 Topología o Red.

Una topología puede ser definida como una lista de relaciones espaciales. Crear y almacenar relaciones hace el almacenamiento más eficiente, así como el procesamiento de datos se hace más rápido. La topología facilita funciones analíticas, la identificación y manipulación de objetos adyacentes o superpuestos. Con la topología podemos definir 2 tipos de asociaciones: La conectividad y la adyacencia. La topología permite definir, como ya se ha visto, a las líneas y polígonos de un mapa.

Las topologías, al igual que los anteriores elementos gráficos podrán ser almacenados dentro de un BD, tanto su relación geográfica y geométrica, como otros atributos (como por ejemplo los costos en tiempo en una red vial).

2.2.7.2 *Base de Datos.*

La característica esencial de cualquier sistema de almacenamiento de datos, como lo son las bases de datos, es que la información debe de estar disponible de manera rápida para ser consultada y/o para combinarse con otra información.

Las bases de datos son el componente de un GIS que nos permite conservar y procesar la información, no se pretende profundizar mucho en esta característica, por lo que se dará solamente una visión global de las bases de datos.

2.2.7.2.1 *Archivos y Acceso a Datos*

Una de las formas más simples de almacenar datos es como una lista de ítems, y para cada adición, los datos nuevos son puestos al final de la lista, por lo que agregar datos es muy sencillo, pero hacer una búsqueda de un ítem específico es muy ineficiente, pues se recorre el archivo de manera secuencial desde el principio hasta encontrar lo que buscamos. Para una lista que contiene n ítems, se toma en promedio $(n+1)/2$ operaciones de búsqueda de un ítem específico. Por ejemplo, en una lista de 10, 000 ítems, donde la operación de búsqueda (lectura y comparación con el que buscamos) toma 0.1 segundo, tomaría en promedio un poco menos de 10 minutos.

Las palabras en los diccionarios, o las guías telefónicas son ordenados alfabéticamente, por lo que la adición de un ítem provocaría que se cree un espacio extra para la información en cualquier lugar del archivo, pero las operaciones de búsqueda son mucho más rápidas. Es muy común, en archivos secuenciales ordenados el hacer búsquedas de tipo binarias. La búsqueda comienza a la mitad del archivo, si el registro buscado es ese, la búsqueda termina, pero si no se hace una prueba para obtener la mitad que contiene el elemento buscado, y se procede a hacer lo mismo con esa mitad. Las búsquedas binarias requieren de $\log_2 (n+1)$ operaciones de búsqueda, por lo que si aplicamos el ejemplo anterior nos tomaría un poco más de un segundo el encontrar el ítem buscado.

En las listas secuenciales y ordenadas, se requiere que los elementos estén ordenados de acuerdo a un atributo clave, que muchas veces o no se posee o no es único (se necesitan búsquedas con varios atributos del ítem), por lo que al archivo se le crean índices, de manera similar a como se le hace a un libro, por lo que la búsqueda se convierte en una operación de 2 pasos: La búsqueda en el índice que nos indica la posición del ítem buscado en la lista, y la lectura de esa segunda lista. Lo complicado de este método es el mantener los índices, pues las operaciones de adición y borrado de ellos obligan a su modificación.

2.2.7.2.2 *Estructura de Bases de Datos.*

Una base de datos consiste en un conjunto de datos almacenados en muchos archivos. En función de poder consultar esta información de uno o más archivos de manera rápida, es necesario que se tenga algún tipo de organización.

Esta organización ha ido evolucionando, y ha pasado por 3 estadios: las estructuras jerárquicas, que resultaron muy poco flexibles y redundantes; las estructuras de red, que resolvieron el problema de la flexibilidad pero no el de la redundancia; y actualmente se utiliza el modelo relacional, que aunque todavía quedan problemas que resolver, ha demostrado la manera más rápida y sencilla de almacenar datos.

Las bases de datos relacionales se olvidan de los apuntadores, demasiado complejos de mantener, y en su lugar almacenan los datos en simples registros almacenados en tablas de 2 dimensiones, estas tablas se relacionan a través de un campo clave redundante, pero que permite una gran flexibilidad y permite además resolver todas las preguntas que se formulen usando las reglas de la lógica booleana y de operaciones matemáticas.

2.2.7.3 *Software para análisis.*

La mayor diferencia entre lo los SIG y la cartografía asistida por computadora, radica precisamente en la capacidad de los primeros de transformar la información geográfica original para responder a preguntas específicas.

Los análisis de un SIG involucran la manipulación de atributos gráficos y de datos. En este estadio los SIG son capaces de apoyar fuertemente la toma de decisiones. Con el análisis las relaciones entre diferente información geográfica y sus características de asociación pueden ser medidas y entendidas.

El análisis de la información de los SIG permiten extraer conocimiento cuantitativo y cualitativo de la información almacenada dentro de ellos, el cual puede ser de varios tipos:

1. Modelado de fenómenos humanos y naturales.
 - 1.1. Descripción del fenómeno, su localización, variables vitales, etc.
 - 1.2. Predicciones y simulaciones del tipo ¿qué pasaría si?.
 - 1.3. Planeaciones.
2. Relaciones espaciales entre dos fenómenos geográficos.
 - 2.1. Relaciones de tipo topológicas (redes), como pueden ser distancias, costos, tiempos, etc.
 - 2.2. Relaciones lógicas entre áreas: uniones, intersecciones, complementos y negaciones, de atributos mapeados, como pudiera ser ingresos, pH, servicios, etc.
3. Agrupamientos, clasificaciones y reconocimiento de patrones, distribución de ingreso, crímenes, distribución de especies, localización de recursos naturales.
4. Diagnóstico, monitoreo y control de diferentes fenómenos: asentamientos humanos, áreas dedicadas a agricultura, daños causados por desastres naturales, seguimiento de ciclones, etc.
5. Conversiones de estructuras raster y vector.

6. Interpolaciones y extrapolaciones de información para obtener datos que no se poseen.
7. Detección de errores.
8. Operaciones matemáticas con información localizada geográficamente, sumas, restas, funciones transcendentales, transformaciones, etc.

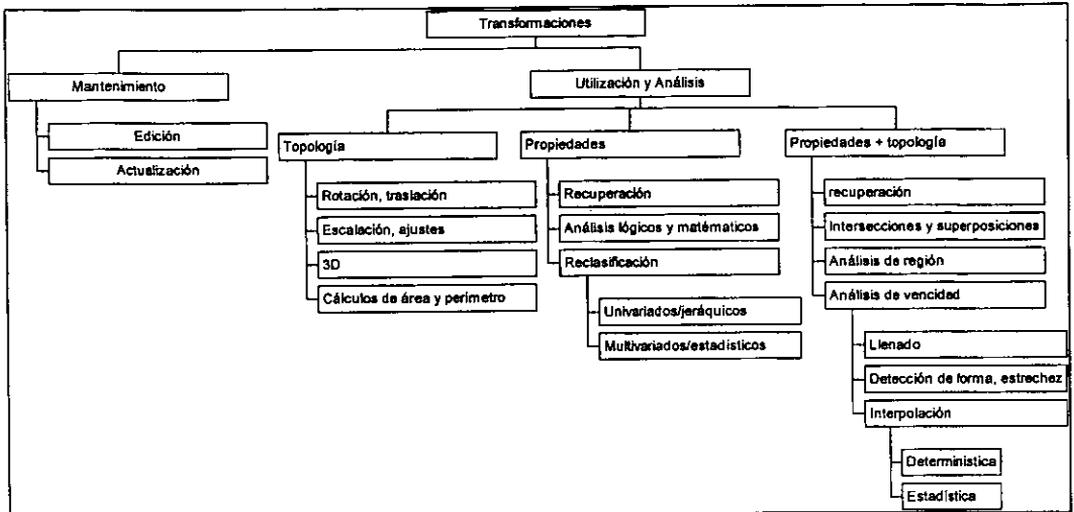


Figura 2.6 Tipo de transformaciones y análisis que puede realizar un SIG.

Los tipos de análisis que un SIG, como se muestran en la figura 2.6, pueden ser desde los más simples, como la recuperación de información dentro de la base de datos, hasta métodos estadísticos uni y multivariados usando funciones de vecindad y de interpolación.

2.3 Alcances y Usos de los SIG.

Como se ha podido ver, la integración de las tecnologías asociadas a la computación con datos espaciales, puede ser una poderosa herramienta para estudios de variados tipos, excediendo las capacidades de los mapas tradicionales, por lo que el alcance de estos sistemas es muy vasto.

Algunos de los usos de este tipo de sistemas son:

- Administración de recursos y ciencias ambientales: Silvicultores y agricultores utilizan los SIG para monitorear, medir, detectar cambios, etc., en los impactos ambientales. Estudios de polución del aire y mares, modelación del clima, localización de recursos de minerales, modelación hidrológica, etc.
- Estudios ecológicos. Los investigadores lo utilizan para estudiar distribución de la flora y fauna, creación de áreas protegidas, cálculos de biodiversidad y riqueza, etc.
- Transportación. Algunos GIS especializados en transporte, utilizan redes (autopistas, calles y caminos, rieles o líneas de autobús) y nodos (paradas de autobús, estaciones, transbordos, intersecciones, etc.) para resolver problemas como la ruta más corta, de distribución, agentes de ventas, control en tiempo real de flotillas, control y planeación del transporte público, sistemas de guía remota, etc.
- Administración pública y gobierno. Autoridades de planeación y desarrollo los utilizan en planeación de usos de suelo, monitoreo del crecimiento de las ciudades, catastro, análisis de crimen, control de enfermedades, desarrollo económico, servicios sociales, asignación de impuestos, creación de distritos electorales, censos, etc.
- Sistemas de respuesta de emergencia. Son utilizados para el trazado de rutas más cortas, localización de centros de atención, localización de áreas de desastre, evaluación de daños por fenómenos naturales y humanos, etc.
- Telecomunicaciones. Las compañías de telecomunicaciones los ocupan para la creación de células en la telefonía móvil; administración, control, mantenimiento y planeación de sus redes; áreas de cobertura, etc.
- Marketing y ventas. Los SIG son utilizados para segmentar los mercados, áreas de influencia de sucursales, redes óptimas de atención, análisis de la competencia, ventas localizadas, localización de clientes, análisis de riesgos en aseguradoras, mercados objetivos, planeación de estrategias de marketing, etc.

- Militares. Queramos o no es una de las actividades humanas, y por supuesto, son las primeras en utilizar las ventajas de los SIG para resolver cuestiones tales como estrategias militares, ubicación de blancos, guiado de tropas, localización de armamento y tropas enemigas, etc.

2.4 Sistemas de SIG Existentes

En la actualidad existe una gran cantidad de aplicaciones GIS, tanto genéricas como de aplicación específica, corriendo en una gran variedad de plataformas y sistemas operativos: en cualquier sistema tipo Unix de grandes mainframes y estaciones de trabajo, como en Linux en PC; en Windows; en VMS, etc.

Los sistemas van desde miles de dólares, como Map Info, Spans, Arc/Info, etc.; , como desarrollos libres como GRASS y GEODYSEY. Desarrollados tanto por empresas privadas y de lucro: Tydac, ESRI, etc., como por instituciones públicas y académicas: como Center for Applied Geographic and Spatial Research, Universidades tales como la de Ohio, Harvard, etc., así como agencias gubernamentales, etc.

Ésta es una pequeña muestra de los sitios en Internet que se pueden encontrar de empresas e instituciones dedicadas a desarrollo de GIS:

• 3DI Geographic Technologies	• IBM's GFIS
• 4thBeach Software	• Clark University
• AI-GEOSTATS GIS	• ILWIS AFRICA
• Able Software	• IntelliMap
• Arc/Info AML FTP site.	• Intergraph Corporation
• ARGUS Census Map USA Page.	• Internet GeoStar
• AutoDesk - AutoCAD Map	• JMapper Inc.
• Avenza	• Magic Software Co.
• AXSES	• Manifold GIS System
• BAGH Technologies -	• ThinkSpace, Inc.
• Bits Per Second	• MAPCOM Systems
• Bentley Systems	• MapInfo Corporation
• Blue Marble Geographics	• Microlmages, Inc. - English Site
• BOSS International	• Northwood Geoscience LTD
• Global Research Labs	• NuGraph Rendering System and PolyTrans
• Caliper Corporation	• OZGis

• Universal Systems Ltd.	• Pacer Infotec Inc.
• CarteGraph	• Paradata Systems Inc.
• Collins Software	• PCI Pacific GeoSolutions
• Compass Informatics, Ltd.	• Pepperwhite Ph.D. Associates, Inc.
• Convergent Group Asia-Pacific	• Planet One GIS Software
• Course Technology	• Professional Geo Systems B.V.
• DataMap Europe Ltd.	• PROSIS S.A.
• Decisionmark Corp.	• RAMTECH Corporation
• DeLorme Mapping	• Safe Software, Inc.
• Digtterra	• The ECDIS Company
• Empower Geographics Inc. Makers of CrossStreets (tm)	• Scan/US®
• Enabling Technology Inc.	• SINTESYS
• Enghouse Systems	• Smallworld Systems Inc.
• The Minnesota Land Management Information Center	• Softree
• ERDAS Inc.	• Solmac Consultancy company
• ER Mapper	• Spatial Innovations
• ESRI - ArcInfo	• SPRING
• ETAK	• SST Development Group
• FieldWorker Products Ltd	• Strategic Mapping Inc.[USA][AD]
• GDS Corporation	• SylvanMaps/OCX(tm)
• Genasys II, Inc.	• Synergos Technologies
• GeoAnalytics, Inc.	• Synoptech Inc
• GeoData Solutions	• System Options Ltd
• Geodyssey Limited of Calgary [CA][COM] Hipparchus GIS Tool	• Tactician Corporation
• Geo InSight International, Inc	• Tekla Oy Technical Information Systems
• GEOsurv Inc.	• Tetrad Computer Applications, Inc.
• GIS Control-Tool Software	• TEXO GIS Software page [IT][COM]
• GIS Software [USA][GOV]	• The GIS Group, LLC
• GIS\Solutions, Inc.	• TopoVista
• GIS Tools Developers of GIS shareware	• USGS
• GRASS GIS home page.	• VISION - From SHL
• Graticule	• WoolleySoft
• Howald Systems Homepage	• xearth
	• Wessex, Inc.

Lista de sitios tomada de la página de gislink: <http://www.gislinx.com/>

3 Área de Estudio: Biogeografía

3.1 Biogeografía.

La Biogeografía es el estudio de la distribución de los organismos, la historia de cómo se forma un arreglo particular y un análisis de las posibles causas de ese arreglo (Simberloff, 1983). La distribución de la biota ha sido tema de gran interés científico, económico y social a lo largo de la historia, incluso antes que apareciera la ciencia como tal. Durante mucho tiempo, el campo de la Biogeografía se desarrolló con la idea de que si los animales y las plantas se encontraban en un área geográfica determinada, tal distribución solo podría explicarse por las condiciones físicas de su entorno. El descubrimiento del Nuevo Mundo y otras áreas geográficas tropicales puso en evidencia que las áreas se pueden caracterizar por grupos de seres vivos exclusivos a ellas (endemitas), aún cuando las condiciones físicas fueran equivalentes entre dos áreas de endemismo o regiones bióticas, esto es, en áreas distintas hay especies distintas (Llorente, 1996).

Una de las clasificaciones más usadas para esta ciencia es aquella que la divide en "histórica" y "ecológica". El objetivo de la Biogeografía histórica es mostrar y explicar la relación de biotas y su conexión histórica a nivel de escalas geológicas. La Biogeografía ecológica considera procesos en una escala de tiempo menor, y precisamente, explica la distribución de especies mediante factores ecológicos y no geológicos (Figura 3.1).

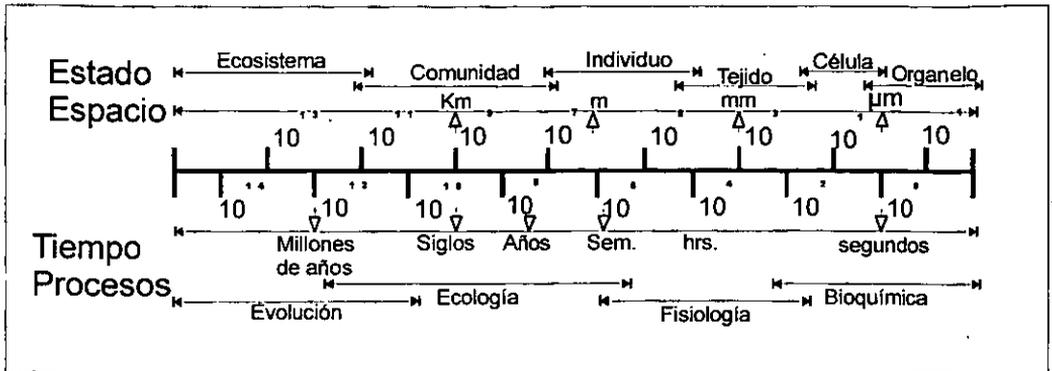


Figura 3.1 Procesos naturales que se dan en un continuo de escalas espacio-temporales.

La Biogeografía se encuentra actualmente en un estado de constante cambio y evolución, y debido a la gran cantidad de datos, ésta requiere de métodos cuantitativos y análisis asistidos por computadora. La Biogeografía cuantitativa es un conjunto de técnicas de análisis cuantitativos que apoyan a la Biogeografía; estos análisis pueden ser análisis mono y multivariados, además de técnicas gráficas que frecuentemente requieren de computadoras para su realización.

Además de la potencia que imprimen las nuevas tecnologías al desarrollo de la Biogeografía existe otro factor (no tan alentador) que le está dando un fuerte impulso a esta ciencia y a las técnicas que utiliza. Dicho factor se refiere a la evidente disminución de la Biodiversidad como consecuencia de la acelerada destrucción de hábitats. La magnitud a la que ésta ocurriendo este proceso no lo sabemos con precisión porque, para empezar, no tenemos un catálogo completo, o al menos adecuado, de dicha biodiversidad.

Una forma convencional de asomarse al problema, es revisar las bases de datos que algunas agencias tienen con respecto a especies extintas o amenazadas de extinción. Por ejemplo el Centro Mundial de Sondeos sobre Conservación, consigna una lista de 4, 589 casos de animales amenazados (IUCN, 1988). La Unidad de Plantas

Amenazadas lista un total de 50, 000 especies de plantas con algún grado de riesgo. Es claro, no obstante, que estas listas rojas son imprecisas e incompletas, pero aún siendo muy burdas, sugieren que estamos frente a un proceso de extinción de gran magnitud.

La explicación de las reducciones contemporáneas (léase inducidas por el hombre) de la biodiversidad, incluye una compleja trama de causas proximales y últimas, determinadas por problemas interconectados, de tipos tan variados como político-económicos (por ejemplo demandas alógenas y distribución desigual de los recursos), sociológicos (tendencias demográficas), ecológicos (no planeación o planeación errónea del uso del suelo), etc. Por ejemplo la deforestación tropical se está dando a un ritmo de 0.7 a 1 % anual, esto equivale a una destrucción anual de un área equivalente al tamaño de Costa Rica, dicho ritmo de destrucción hace que se estime una caída de biodiversidad tropical de 4 a 6 mil especies extintas al año (Dirzo, 1990).

Por las razones anteriormente mencionadas se deben tomar medidas urgentes que permitan frenar ese ritmo y que sean compatibles con los intereses de desarrollo. Una manera de afrontar la "crisis de la biodiversidad" es mediante la designación de áreas protegidas. El Banco Mundial cuenta con un fondo para apoyar proyectos como los siguientes:

- Creación de áreas protegidas.
- Manejo de áreas protegidas existentes.
- Esfuerzos para conservar la biodiversidad existente fuera de las áreas protegidas.
- Conservación *ex-situ*.
- Estudios, investigación y monitoreo.
- Estrategias, planeación, políticas e instituciones.

Aunque más del 5% de la superficie de la Tierra está designado a áreas protegidas (World Bank, 1995) éstas no garantizan la conservación de un buen porcentaje de la biodiversidad del planeta.

El primer paso para el diseño y manejo de áreas protegidas es describir, inventariar y mapear la biodiversidad, pues es crítica la localización exacta de las reservas para conservar el máximo número de especies. Así, el trabajo de conservación depende de los sistemáticos y de los biogeógrafos, ya que es indispensable conocer dónde establecer las reservas y localizar los sitios de endemismo y diversidad (Soule, 1990).

3.2 Biogeografía Cuantitativa

El análisis cuantitativo de datos biogeográficos es sólo un paso dentro de un estudio biogeográfico, pues antes de éste se deben tener claros los objetivos del estudio y se debe realizar una serie de decisiones en cuanto a la estructura de los datos a ser analizados (Crovello, 1981). La misma recopilación e integración de datos es un paso previo al análisis.

La biogeografía cuantitativa y en sí la Biología poseen algunos conceptos, que bien valdría la pena definirlos para poder trabajar con ellos, que permitan comunicarnos con el experto en el área. Como en la parte de los conceptos de SIG, no se busca ser un tratado extenso y formal sobre dichos tema.

3.2.1 Especie.

Las especies, generalmente son la unidad de biodiversidad y conservación, por eso es importante que podamos conocer qué se entiende por ese concepto (Claridge, 1997).

Los científicos tienen el ideal que sus conceptos sean lo más posible teóricos, generales y aplicables. Desafortunadamente, esas tres metas tienden a crear conflictos y un ejemplo de ello es el concepto de especie en la Biología.

La controversia en torno al concepto de especie es una de las más ricas en el ámbito de la biología (Claridge, 1997). Es una de las discusiones que tiene amplias zonas de contacto entre la filosofía y la práctica, se tomarán las siguientes definiciones de especie como las más apropiadas para la Biogeografía ecológica y cuantitativa:

Especie biológica

Grupo de poblaciones naturales que se entrecruzan y que está aislado reproductivamente de otros grupos similares. (Mayer, 1969)

Especie ecológica

Linaje o conjunto de linajes afines, que evoluciona separadamente de otros y que ocupa una determinada zona adaptativa. (Van Valen, 1976)

En el caso de la definición de especie biológica se centra en los mecanismos de especiación, como por ejemplo la inhabilidad para tener descendencia como el factor más importante; cuyo concepto es difícil de aplicar en plantas, donde la hibridación es un proceso común. En cambio el concepto ecológico considera parámetros físicos y las propiedades de los organismos (zona adaptativa) para adaptarse al medio.

En nuestro caso (desarrollo de un SIG para la Biogeografía), el concepto de especie se tomará como una de las dos unidades en los estudios biogeográficos (como se verá más adelante la otra unidad es geográfica) las cuales son denominadas Unidades Taxonómicas Operativas (OTUs), dejándose a libertad del investigador el concepto de especie que más le convenga, así como el tipo de clasificación que desee, aunque se recalca que es muy importante que en los trabajos sistemáticos se indique explícitamente el concepto de especie aplicado, pues diferentes conceptos de especie pueden resultar en diferencias en las estimaciones de la diversidad específica, en el listado de especies endémicas, etc.

3.2.2 Definición del Área de Estudio.

En un estudio biogeográfico es importante definir el contexto, es decir, los objetivos, los grupos taxonómicos a incluir en el análisis, así como la propia dimensión geográfica, siendo esta última lo que se denomina "área de estudio".

Se asume que antes de proceder a la colecta de la información que será analizada, el trabajo ha sido ubicado dentro de un marco de referencia general que incluye los objetivos y metas que pretende cumplir, así como los factores que afectan a la investigación en sí, como lo son los recursos humanos y económicos disponibles.

3.2.3 Definición OGU's.

Varias de las técnicas para realizar análisis biogeográficos se basan en una generalización geográfica de los registros de las recolectas de los organismos, es decir, se supone una distribución uniforme en un área dada de los organismos colectados en un punto (Unidad Geográfica UG. Figura 3.2).

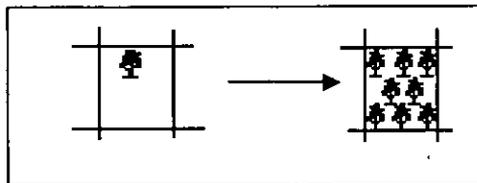


Figura 3.2 Generalización de una unidad geográfica: se considera que la especie recolectada en un punto, se encuentra distribuida uniformemente en el área de la unidad geográfica.

En la Biogeografía cuantitativa se analizan relaciones a partir de unidades geográficas, estas unidades son llamadas Unidades Geográficas Operativas (OGU). Las unidades geográficas así definidas son las unidades mínimas de análisis, representan el nivel máximo de resolución, por lo que pueden agruparse mas no desagregarse.

Generalmente son definidas como cuadros de tamaño regular, ya que permite obtener unidades con áreas de igual tamaño, cuya validez de su comparación en

cuanto al número de especies es mayor que entre áreas de diferentes tamaños, pues en general, el número de especies aumenta con el tamaño del área.

3.2.4 Matriz de Presencia-Ausencia.

La matriz de presencia-ausencia consiste en un arreglo bidimensional donde se registran la existencia de especies en las OGU. Aunque puede parecer lógico que datos de presencia-ausencia sean inferiores a datos cuantitativos (abundancia de la especie en las OGU), se ha observado bajo diferentes circunstancias, que los datos binarios pueden producir resultados tan buenos o aún mejores que los datos no binarios (Green, 1979). Esto puede ser debido, al menos parcialmente, al hecho de que las presencias y ausencias tienden a ser más robustas ante los errores inevitables del muestreo.

3.2.5 Índices de Diversidad.

Existen varias propuestas para definir y cuantificar la diversidad. Con fines de esquematizar algunas propuestas de cuantificación de la diversidad se utiliza la siguiente clasificación:

α (diversidad alfa): Se asocia al número de especies en una comunidad, técnicamente para una OGU se tiene la diversidad alfa, y para grandes áreas (técnicamente el área de estudio), se le denomina diversidad gamma (γ).

β (diversidad beta): Es un grado de heterogeneidad biológica en el área de estudio, si la mayoría de las especies en las OGU del Área de Estudio son las mismas, la diversidad beta es baja.

3.2.6 Estimadores de Riqueza.

Generalmente, las recolecciones en campo son muestras aleatorias e incompletas, pues hay especies raras y difíciles de encontrar, que muy probablemente no se tenga ningún ejemplar en la recolección, además de que las recolecciones

pueden estar cargadas de factores espurios (preferencias del recolector, recursos destinados a la tarea, etc.) que hagan la muestra no sea de todo válida en cuanto al número y tipo de especies que se encuentran en el área, por lo se hace evidente la estimación de la riqueza. Ronald Fisher es el primero en estimar de forma teórica la relación que debería tener el número de especies recolectadas y el área de muestreo, su modelo era capaz de describir la cantidad de especies a hallar en la muestra, como una función de la cantidad de individuos que tiene cada especie en la muestra, y, sumando los individuos de todas las especies, como una función del tamaño total de la muestra.

A la relación de número de especies recolectadas y el área de muestreo se le conoce como la "relación especie-área". El interés radica no sólo en problemas estadísticos de muestreo sino que es de capital importancia en problemas de biogeografía y conservación pues permite, entre otras cosas, evaluar el tamaño de una reserva natural y entender la dinámica de extinción de especies en islas y hábitats fragmentados. Un adecuado conocimiento en las relaciones especie-área, resulta de gran importancia en los estudios de evaluación y de inventario de la riqueza biológica de una determinada región.

Por lo que los estimadores de riqueza son funciones estadísticas que permiten calcular un nuevo valor de la diversidad gamma a partir de los valores de α de las OGU's ya sea mediante tendencias de tipo inductivo, o mediante el establecimiento de relaciones y proporciones.

3.2.7 Distribución Geográfica y Lista de Especies.

Este tipo de análisis permiten al biogeógrafo observar la distribución de especies en términos de OGU's y no simplemente de recolectas, como una generalización que le facilitará la generación de hipótesis.

La distribución y las listas de especies darán un cierto grado exploratorio de los datos permitiendo que el investigador se familiarice con ellos y le darán elementos para emitir juicios sobre los límites y alcances de los análisis, así como sus resultados.

3.2.8 Similitud Biogeográfica.

La similitud biogeográfica se mide a través de la semejanza entre los componentes bióticos. Mientras mayor porcentaje de especies en común contiene dos áreas, mayor es su similitud biogeográfica.

Para la medición de la similitud se utilizan los denominados índices de similitud, que no es más que una manera de medir la semejanza entre dos conjuntos de especies contenidas en sus respectivas OGUs, permitiendo así medir la semejanza entre dos regiones o áreas geográficas en función de las especies que contienen y pueden o no tomar en cuenta los valores de abundancia de cada una de las especies. Los valores que toma la similitud están en el rango de 0 a 100% (0 a 1).

3.2.9 Matriz de Similitud.

Los índices de similitud sirven para establecer una medida de semejanza entre dos OGUs; la matriz de similitud contiene las comparaciones entre todos los pares posibles de OGUs. Los encabezados de las columnas y filas de la matriz de similitud son cada una de las OGUs, y los valores contenidos dentro de la matriz son precisamente las medidas de similitud. Con una breve exploración se puede ver que la matriz es simétrica, con la diagonal principal formada por unos, debido a que es la comparación de la OGU contra sí misma. La importancia de esta matriz es que permite realizar otro tipo de análisis, como los análisis de agrupamiento.

3.2.10 Análisis de Agrupamiento.

Los métodos de agrupamiento buscan formar regiones que poseen un grado de semejanza determinado (utilizando un índice de similitud para cuantificar dicha

semejanza entre las OGU's). Los métodos de agrupamiento se pueden clasificar en divisivos y aglomerativos.

3.2.10.1 Métodos Aglomerativos.

Dentro de este tipo de métodos los más comunes son los secuenciales jerárquicos, dentro de los que se encuentran el ligamiento simple, el ligamiento completo y el UPGMA, los cuales forman un fenograma, que es un tipo de gráfica dirigida (árbol).

El algoritmo inicia eligiendo el par de OGU's cuya similitud es la mayor de entre los pares posibles, la matriz de similitud es recalculada considerando ahora como una unidad al par de OGU's elegidos; la matriz de resultado es de un renglón y una columna menos y se ha formado parte del fenograma: dos hojas unidas a un determinado nivel de similitud. En cada paso la matriz de similitud es recalculada reduciéndose en un renglón y en una columna; el algoritmo termina cuando todas las OGU's han sido ubicadas en el fenograma.

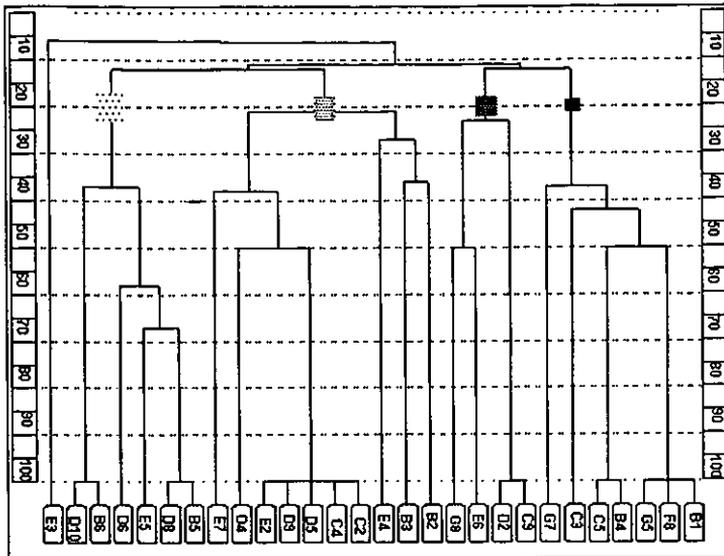


Figura 3.3 Ejemplo de un fenograma, con un nivel crítico de 20%.

La manera en que se recalcula la matriz de similitud hace la diferencia entre los tipos de ligamiento simple, completo o promedio (UPGMA).

El fenograma, puede “partirse” a un determinado “nivel crítico” para conformar grupos de OGUs con características homogéneas (Figura 3.3).

3.2.10.2 Métodos Divisivos.

La estrategia de los métodos divisivos es iniciar formando dos grupos a partir del conjunto total de datos y repetir esto para cada uno de los dos grupos y así sucesivamente hasta llegar a la unidad, es decir, hasta que el grupo contenga una sola OGU.

El criterio para dividir un grupo en dos generalmente depende de una función de optimización. Así, se exploran varias combinaciones de los posibles grupos y se elige la que cumpla mejor con el criterio de optimización.

3.2.11 Definición de Áreas Prioritarias para la Conservación.

La definición de áreas prioritarias para la conservación es un tema que actualmente ocupa a muchos investigadores. Se han propuesto varios métodos para su definición, que se pueden clasificar en dos grupos, los taxonómicos y los cladistas. Los primeros intentan definir áreas geográficas a proteger con base en los listados florísticos o funísticos y capturar el mayor número de especies. Los cladistas toman atención a la filogenia de los organismos a proteger con la finalidad de proteger la mayor heterogeneidad genética.

3.2.12 Jerarquía y Escala.

Los sistemas naturales exhiben una organización jerárquica con patrones anidados y procesos que ocurren en un amplio rango de escalas espacio-temporales (Miller, 1994), por lo que la escala debe de ser considerada al realizar análisis sobre biodiversidad. Por ejemplo, a una determinada escala un proceso puede parecer homogéneo, mientras que a otras puede parecer heterogéneo.

La palabra escala se utiliza para referirse a determinados aspectos del espacio y del tiempo, por ejemplo, tamaño del área o longitud del tiempo (Turner & Gardner 1991). La teoría de jerarquías (Salthe, 1985) estudia las interacciones de los fenómenos entre diferentes escalas y está orientada hacia las "entidades", es decir, los límites de las entidades individuales son importantes. Se entiende por entidad a una representación teórica de una cosa como un individuo. En algunos casos se asume que la variación espacial y temporal están correlacionadas, sin embargo, esto no es una regla general. Por ejemplo, el impacto del hombre sobre la naturaleza a través de períodos de tiempo muy cortos (Hengeveld, 1990:17).

La jerarquía es una estructura que se caracteriza por tener un nivel de organización ocupada por entidades de diferentes escalas y la escala es una

graduación relativa basada en el tamaño y ámbito espacial de influencia, en ocasiones reflejado en la duración del tiempo del fenómeno (Salthe, 1985).

3.2.12.1 Jerarquía en Biogeografía

La región biogeográfica es un nivel de organización en la jerarquía ecológica que surge de la interacción de ecosistemas locales y su ubicación dentro de esa jerarquía es debajo del nivel de *superficie de la tierra* y arriba del nivel de *ecosistema* (Salthe, 1985).

En biogeografía histórica, la jerarquía expresa relaciones ancestrales, mientras que en los métodos fenéticos, como el análisis de agrupamiento, la jerarquía de los fenogramas simplemente son un artefacto del método pero no expresa al fenómeno natural. La jerarquía en un fenograma es útil cuando se decide una línea de "nivel crítico" que corta al fenograma y que define las regiones biogeográficas.

3.2.12.2 La Escala Biogeográfica

Las inferencias estadísticas son sensibles a la escala espacial. Cuando se realizan regresiones a partir de datos espaciales es muy importante considerar la escala, ya que los parámetros de los modelos así obtenidos no serán comparables (Wong, 1996). Se han intentado plantear varias estrategias estadísticas para detectar la escala adecuada para estudiar los patrones de distribución de los organismos (Turner, 1991). De hecho, el problema del "tamaño del cuadro" en análisis biogeográficos es un problema de escalas, al cual también se le han propuesto soluciones de tipo estadístico (Phipps, 1975). Anderson y Marcus (1993) muestran que el tamaño de cuadro afecta significativamente la estimación de la densidad de especies y opinan que no hay un tamaño de cuadro que intrínsecamente sea mejor a otros.

Varias medidas relacionadas con la biodiversidad son sensibles al tamaño de la muestra, y esto hace que también puedan serlo al cambiar de escala espacial de análisis. Por ejemplo, casi todos los índices de similitud son sensibles al tamaño de la

muestra (Wolda, 1981), lo que también genera un problema de compatibilidad entre escalas al utilizar este tipo de herramientas. También los índices de rareza son sensibles al tamaño de la muestra (Gotelli & Graves, 1997), razón por la cual también debe tenerse precaución al manejarlos a diferentes escalas espaciales. Las mediciones de una misma variable a diferentes escalas puede llevar a conjeturas contrarias. Por ejemplo, puede deducirse que los topos de la alfalfa son una plaga que afecta a los cultivos cuando se analizan a escala individual y que es la idea preponderante en la literatura sobre agricultura, pero se puede observar que son benéficos e incrementan la productividad cuando se analizan al nivel de población, ya que modifican algunas condiciones en el suelo que producen beneficios al cultivo (Geng, 1997).

3.2.13 El Problema de la Unidad de Área Modificable

El problema de la unidad de área modificable (MAUP por sus siglas en inglés) se refiere a la inconsistencia de los resultados de los análisis de un mismo conjunto de datos agrupados espacialmente de diferente manera (Wong, 1996). A un nivel de agregación de las unidades geográficas originales de datos pueden observarse algunas correlaciones mientras que a otras no.

El MAUP consiste de dos subproblemas: el efecto de escala y el de zonificación (Wong, 1996). El primero se refiere a la inconsistencia de los resultados estadísticos del análisis de los mismos datos a diferentes niveles de agregación, por ejemplo, raster a diferente resolución. El efecto de zonificación se refiere a inconsistencias al agregar los datos a resoluciones similares pero en diferente forma, por ejemplo, un raster de cuadros vs. un raster de hexágonos.

Cuando las unidades adyacentes se unen, para formar una unidad mayor, en la que el atributo es calculado como el promedio, se disminuyen las diferencias y entonces la varianza para el área total decrece. Además, como la similitud entre las

unidades de área no es homogénea en toda la región y en todas las direcciones, al agregar de forma diferente la varianza también se verá afectada.

3.3 Algunas Consideraciones:

Es importante tener en cuenta que la biología no ha alcanzado el nivel de precisión de las ciencias exactas y que durante la realización de cualquier investigación, el biólogo debe de asumir compromisos que le dan una cierta dosis de inevitable subjetividad al trabajo. Es por esto, que los usuarios de cualquier herramienta tecnológica, similar o no a ésta, comprenda no los fundamentos matemáticos de las técnicas de análisis, pero sí sus alcances, suposiciones, ventajas y desventajas. Pues si bien las computadoras y los análisis cuantitativos pueden potenciar la mente humana, no podrán reemplazarla.

También es importante recalcar que el problema de la diversidad biológica es directamente relevante a México. Este país es uno de los más importantes centros de biodiversidad del planeta, dentro de los 12 países más ricos en biodiversidad del planeta, y al mismo tiempo un "hot spot" por sus formidables problemas de conservación (Dirzo, 1998).

4 Discusión de la Metodología a Seguir

Hay una leyenda que dice que el primer analista de sistemas apareció en escena hace 6000 años, durante la construcción de las pirámides de Egipto. Este personaje preocupado por los ineficaces métodos utilizados para construir los magníficos monumentos, hizo la siguiente aclaración a Kufu, el constructor de la Gran Pirámide de Keops :

“Oh, noble Kufu, es tiempo que nos organicemos. Hemos empujado esta roca a través del desierto en dirección equivocada durante siete años. Lo que necesitamos es una Técnica de Rutas y de Levantamiento de Pirámides” Según los rumores, fue azotado en la plaza pública y nunca más se supo de él... al menos hasta el siglo XX. (Kenniston W, James Steiner,1978).

El primer analista de sistemas intentó idear un modo mejor de construir un producto. Los analistas de sistemas de hoy en día también intentan hallar mejores vías para crear nuevos productos, como son los Sistemas de Información (SI).

Construir cualquier sistema no es trivial es una tarea compleja y a menudo abrumadora, es por eso que la construcción de cualquier SI requiere de una metodología para hacerse, con la finalidad de que todos los esfuerzos permitan llegar al objetivo. Por lo que las metodologías de desarrollo de software permiten:

- Tener una concepto claro y limitado de lo que el software va a hacer.
- Permiten planear su desarrollo:
 - Cálculo de tiempo.
 - Recursos a utilizar (computadoras, software de desarrollo, etc.).
 - Recurso humanos necesarios.
- Evitan los riesgos del desarrollo:
 - Tiempos excesivamente largos en la creación del software.

- Aumento innecesario de la complejidad.
- Elevación del costo.
- No cumple con los requerimientos pedidos.
- Interfaz confusa, poco coherente, desempeño pobre, inestabilidad, etc.

Un SI puede considerársele como un ente abstracto que de alguna manera tiene vida, es decir, tiene un nacimiento, crece a través de las modificaciones y mejoras que se le hacen para mantenerlo al día, e inclusive muere si ya no cumple con los objetivos o es demasiado costoso modificarlo (técnica y económicamente hablando), también muere cuando sus objetivos ya han desaparecido o es simplemente sustituido por un sistema nuevo.

4.1 Ciclo de Vida de un Sistema de Información (SI).

El ciclo de vida de un SI es un proceso por el cual los analistas de sistemas, los ingenieros de software, los programadores y los usuarios finales elaboran sistemas de información y aplicaciones informáticas.

Una metodología es una versión amplia y detallada de un ciclo de vida completo del desarrollo de sistemas que incluye:

1. Tareas paso a paso para cada fase.
2. Funciones individuales y en grupo desempeñadas en cada tarea.
3. Productos resultantes y normas de calidad para cada tarea.
4. Técnicas de desarrollo.

Una metodología o modelo de ciclo de vida es un modelo prescriptivo de lo que pasaría entre la primera chispa y el último aliento (de aquí en adelante se referirá de manera indistinta al ciclo de vida como el modelo de ciclo de vida o como una metodología de ciclo de vida). Es importante saber los requisitos del software a desarrollar (magnitud, software empotrado, de aplicación general, científico, a la medida, etc.), los recursos humanos con que se cuentan y la experiencia de éstos, y el tiempo y dinero disponibles para poder seleccionar el ciclo de vida más apropiado.

4.1.1 Cascada Pura.

Algunos autores proponen como las partes de las que consta el desarrollo de un SI como lo siguiente:

1. **Planificación:** Se propone este paso en el caso de empresas e instituciones, como la forma de generar un plan de sistemas que tenga una visión general de la misión de la organización, con la finalidad de identificar y establecer prioridades sobre aquellas aplicaciones de los sistemas de información cuyo desarrollo reporte máximos beneficios para la organización.

En nuestro caso, debido a que nuestro sistema no va a ser diseñado con el objetivo de resolver las necesidades específicas de una organización, sino de muchas, es decir, un software comercial, este paso casi no aplica en nuestro desarrollo y solo nos permitirá detectar la necesidad del sistema.

2. **Análisis:** El análisis de sistemas es el primer paso dado clásicamente hacia la construcción de una aplicación de SI. En esta fase se hará el estudio del entorno del problema y la subsiguiente definición y establecimiento de prioridades entre las necesidades planteadas con el fin de resolver el problema.

En este paso se pueden generar modelos e inclusive prototipos que nos permitan tener una mejor visión de las entradas, procesos y salidas que se debieran tener para cubrir las necesidades.

3. **Diseño:** Es la evaluación de las diferentes alternativas, así como la especificación detallada de la solución final. El primer tipo de diseño debe de ser general, de manera que permita tener un esquema del diseño global, de ahí se generará un diseño detallado en el que se centran las especificaciones detalladas de los componentes de dicho esquema.

El diseño detallado se desglosa en dos partes: la primera se refiere al diseño externo el cual es el conjunto de especificación de la interfaz del sistema con sus usuarios. La segunda parte se refiere al diseño interno el cual es el conjunto de especificación del software del sistema en su estructura y lógica.

4. Implantación: es la construcción del nuevo sistema así como sus pruebas e instalación en los equipos y/o plataformas en que va a correr. Durante esta fase es donde propiamente se escribe el código que una vez compilado nos dará la herramienta buscada.

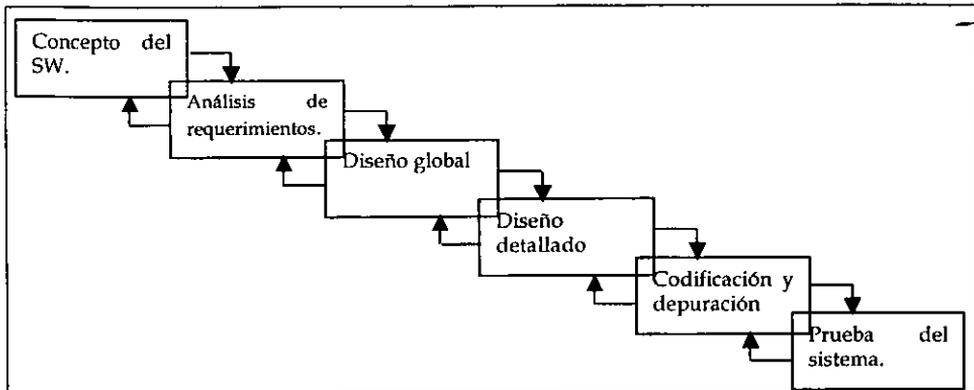


Figura 4.1 Cascada Pura.

La metodología descrita anteriormente corresponde al modelo de cascada pura (figura 4.1). En este modelo un proyecto progresa a través de la secuencia ordenada de pasos de análisis, diseño y desarrollo. En cada paso se hace una revisión al final para determinar si se está preparado para pasar a la siguiente etapa. Cuando dicha revisión determina que el proyecto no está listo para pasar a la siguiente etapa, permanece en la etapa actual hasta que esté preparado.

El modelo de cascada pura está dirigido por documentos; es decir, los productos principales del trabajo que se pasan de etapa a etapa son documentos.

Este modelo se utiliza correctamente para ciclos de productos en los que se tiene una definición estable del producto, y también cuando se está trabajando con técnicas conocidas. Pero tiene el inconveniente de que no proporciona resultados tangibles en forma de software hasta el final del ciclo de vida, además de que cuando se utiliza este modelo, olvidar algo puede suponer un error costoso.

4.1.2 Codificar y Corregir.

Este es un modelo bastante común, el cual puede estarse empleando si no se ha seleccionado un ciclo de vida en forma explícita. No hay ninguna gestión, no hay ninguna planeación ni evaluación de riesgos, solo codificación pura. Para proyectos muy pequeños, programas de demostración y para los prototipos desechables puede ser una opción. Pero para proyectos pequeños, y no digamos medianos y grandes, es demasiado riesgoso, y todo lo que esta palabra implica: costoso, que se entrega tarde o inclusive nunca se entregue, no cumpla los requisitos de calidad (software muy inestable), no cumpla con los objetivos planteados, desempeño deficiente, interfaz poco coherente, etc.

La idea de exponer este modelo, es precisamente el de evitarlo al máximo, y sólo usarlo en casos indicados para ello (prototipos desechables).

4.1.3 Espiral.

El modelo en espiral divide al proyecto de software en miniproyectos. Cada miniproyecto se centra en uno o más riesgos importantes hasta que estén controlados. El concepto riesgo se define ampliamente en este contexto, y puede referirse a requerimientos poco precisos y/o comprensibles, problemas de ejecución importantes, problemas con tecnología subyacente, etc.

Es un modelo iterativo, donde cada iteración supone que el proyecto pasa a una escala superior, donde se comprueba que se tiene lo que se desea, y se empieza a trabajar en el siguiente nivel de la siguiente manera:

1. Determinar objetivos, alternativas y límites.
2. Identificar y resolver riesgos.
3. Generar las entregas de esta iteración, y comprobar que son correctas.

4. Planificar la siguiente iteración.
5. Establecer un enfoque para la siguiente iteración.

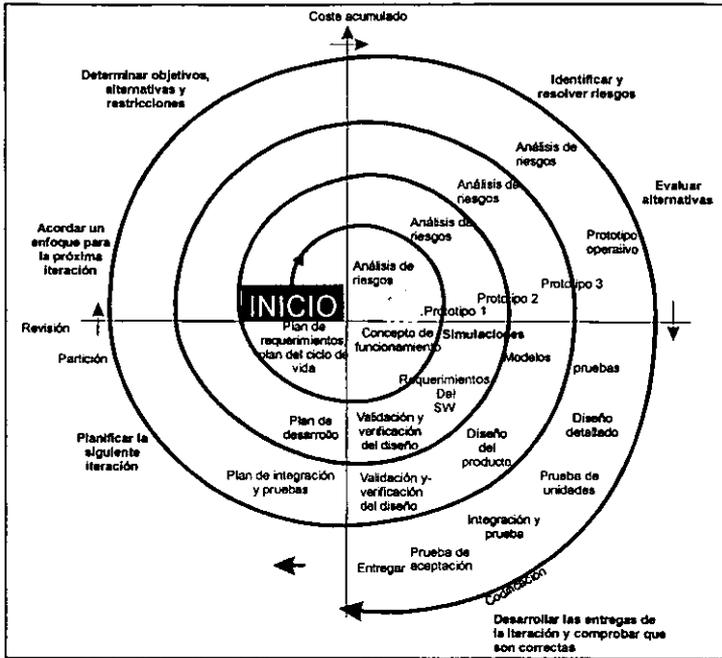


Figura 4.2 Espiral.

Este modelo se puede combinar con otros modelos de ciclo de vida, por ejemplo se puede empezar con una espiral para definir claramente los requerimientos y una vez que éstos se tengan claros seguir con una cascada como modelo de ciclo de vida.

Una ventaja del modelo es que mientras los costos suben, los riesgos disminuyen (el tratar de controlar todas las variables de un proyecto es no controlar ninguna). Aunque este modelo requiere un gestión concienzuda, atenta y que exige conocimientos profundos.

4.1.4 Cascadas Modificadas.

Éstos son en realidad un conjunto de metodologías que parten del de cascada pura, modificando alguna regla y/o característica del modelo de cascada pura. Como por ejemplo el de cascada con fases solapadas, en donde se puede pasar a la siguiente etapa sin haber terminado la anterior, al ser el mismo equipo de trabajo el que empieza y termina, se puede ser un poco más flexible con la documentación.

Otro ejemplo de cascada modificada es el de cascadas con subproyectos, el cual se basa en la premisa de que el desarrollo de los sistemas presentan unas áreas que incluyen sorpresas en el diseño, pero otras no las presentan, entonces ¿Por qué retrasar éstas en función de las primeras, si pueden trabajarse de manera separada?, por lo que este modelo permite dividir el proyecto en subproyectos (Figura 4.3), los cuales cada uno puede proseguir a su propio ritmo. El principal riesgo de este enfoque es la presencia de interdependencias imprevistas.

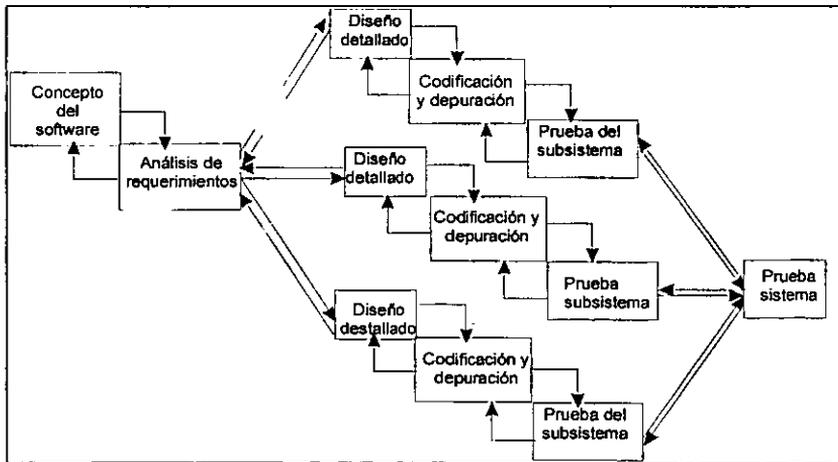


Figura 4.3 Cascada con subproyectos.

4.1.5 Prototipo Evolutivo.

El prototipo evolutivo es el modelo de ciclo de vida en que se desarrolla el concepto del sistema a medida que se avanza el proyecto. Normalmente se comienza con los aspectos más visibles del sistema, los cuales se presentan al cliente y se continúa el desarrollo del prototipo basándose en la retroalimentación que se recibe. En algún punto se decide que el prototipo es lo suficientemente bueno y se entrega como producto final.

Este modelo se utiliza cuando los requerimientos cambian con rapidez, el usuario es reacio a especificar el conjunto de requerimientos, o cuando se quieren probar arquitecturas o algoritmos.

Sus riesgos son que no se puede tener cuantificado el esfuerzo al inicio del proyecto, ni cuándo se tendrá un producto aceptable, además de que es muy fácil que caigamos en el modelo de codificar y corregir. Un prototipo evolutivo real incluye un análisis de requerimientos real, diseño real, en niveles inferiores a las que utilizan los modelos anteriores.

4.1.6 Entregas por Etapas.

Es otro modelo donde el software se muestra al cliente en etapas refinadas sucesivamente. Supone que se conoce exactamente que es lo que se va a construir cuando se procede a construirlo. No se entrega el proyecto al final de un solo golpe, sino que se entrega por etapas.

Este modelo parte de que se hayan completado las siguientes actividades del modelo de cascada (figura 4.4):

1. Concepto del software.
2. El análisis de sus requerimientos.
3. Diseño global.

A partir de aquí se procede a realizar el diseño detallado, la codificación, depuración y prueba dentro de cada etapa, permitiendo entregar al cliente partes

funcionales, que en caso de ser posible, se puedan entregar las prestaciones más importantes al principio, y permiten al cliente poder usar el software desde el principio. El riesgo que se corre con este modelo es el de no tener bien planeado los tiempos, y darnos cuenta que un elemento de la fase 2 no corre hasta que se termine un elemento de la fase 4.

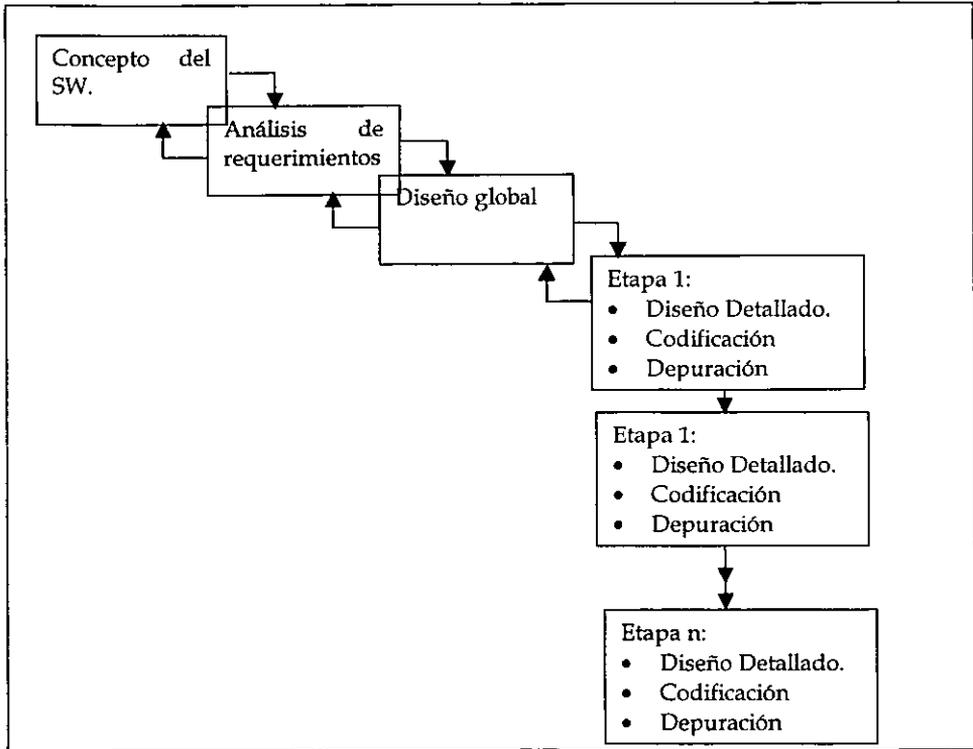


Figura 4.4 Entrega por etapas.

4.1.7 Diseño por Planificación.

Es muy similar al anterior, la diferencia es que se tiene una fecha de entrega inamovible. Por lo que se deberá tener algo para esa fecha.

Como se puede suponer la parte crítica de este modelo es priorizar las prestaciones y planificar sus etapas de tal forma que contenga las prestaciones de mayor prioridad al principio, y las de menor prioridad al final.

4.1.8 Entrega Evolutiva.

La entrega evolutiva es un modelo de ciclo de vida que se encuentra entre el prototipo evolutivo y la entrega por etapas. Se desarrolla una versión, se muestra al cliente, y se refina el producto en función de la retroalimentación.

Las diferencias principales entre el prototipado evolutivo y la entrega evolutiva, son que el énfasis inicial de la entrega evolutiva se pone en el núcleo del sistema y en el prototipado evolutivo se ponen en los aspectos visibles del sistema.

4.2 Paradigmas de Programación.

Una técnica es un método que aplica herramientas y reglas específicas para completar una o más fases del ciclo de vida del desarrollo de sistemas. Uno de sus sinónimos habituales es paradigma.

Los paradigmas son apoyados por los lenguajes de programación. Se dice que un lenguaje apoya a un paradigma si proporciona los recursos que hacen conveniente (razonablemente fácil, seguro y eficiente) utilizar ese paradigma (Stroustrup, 1993). Un lenguaje no apoya una técnica dada si se requiere un esfuerzo o una habilidad excepcionales para escribir tales programas.

4.2.1 Programación Estructurada.

Las técnicas estructuradas son métodos formales de división de un problema en fragmentos y relaciones manejables, y su posterior reunión, nos darán la solución del problema planteado.

La programación estructurada es una técnica orientada a los procesos para el diseño y la escritura de programas con mayor claridad y consistencia.

Esencialmente este paradigma tiene dos vertientes:

1. Programación por procedimientos: Decidir qué procedimientos se desean utilizar y los mejores algoritmos que se puedan encontrar.
2. Programación modular: Decidir en qué módulos se desea dividir el programa de modo que los datos queden ocultos entre módulos.

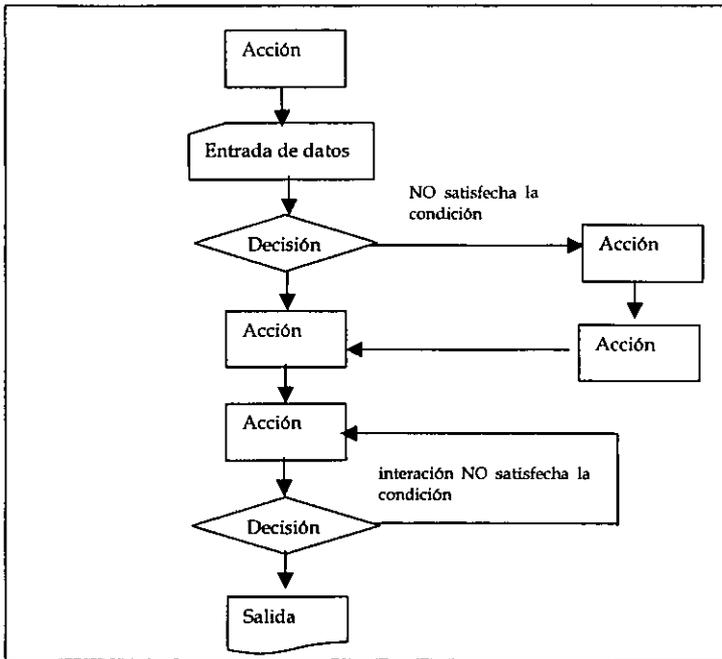


Figura 4.5 Programación estructurada.

4.2.2 Programación Orientada a Objetos:

La programación orientada a objetos es un paradigma reciente que, de manera sencilla, busca algo más que la creación de programas escritos basándose en procedimientos y organizados de forma lógica. El fin de las técnicas orientadas a objetos es el de facilitar la programación, hacerla más efectiva, mediante la escritura

de software que trate con cosas del mundo real. Los métodos de programación orientados a objetos reconocen que la mayoría del software intenta modelar o hacer de forma virtual objetos con los que trabajamos en la vida diaria. Más que reducir esto a un conjunto de reglas y procedimientos que realicen esta tarea, los lenguajes orientados a objetos permiten a los desarrolladores una expresión mejor de dichos objetos directamente en el código. Es de esperar, que éstos conduzcan a un código más expresivo que sea más sencillo de desarrollar y menos costoso de mantener. Para lograr estos objetivos el paradigma expone tres ideas principales que cualquier lenguaje de programación orientado a objetos debe apoyar:

- Encapsulamiento: Capacidad para ocultar los detalles de la implementación de los objetos. Esto significa que la implementación interna del objeto en realidad no es asunto de nadie salvo de él mismo. El objeto también ofrece una interfaz pública a través de la cual se puede involucrar; esta interfaz constituye un compromiso entre los datos y sus usuarios.
- Herencia. La herencia es el medio para lograr la reusabilidad de código dentro de una aplicación, mediante la creación de objetos que heredan la funcionalidad de otros objetos. Se utiliza cuando se posee una clase existente que tiene una funcionalidad básica que se desea, pero no se comporta exactamente como se pretende. La clase heredada será como un superconjunto de la clase base, permitiendo la reutilización de la funcionalidad disponible de la clase base, al mismo tiempo que se le añade una nueva funcionalidad.
- Polimorfismo: Capacidad del código para mostrar múltiples comportamientos, lo que permite adaptar una amplia variedad de situaciones que inicialmente no se habían planificado. El único inconveniente de esta característica es la sobrecarga que impone al equipo que corre el programa (debido a que el compilador no sabe en tiempo de diseño el objeto que

utilizará, por lo que se hará esta decisión en tiempo de ejecución, lo que impone una mayor carga de cómputo).

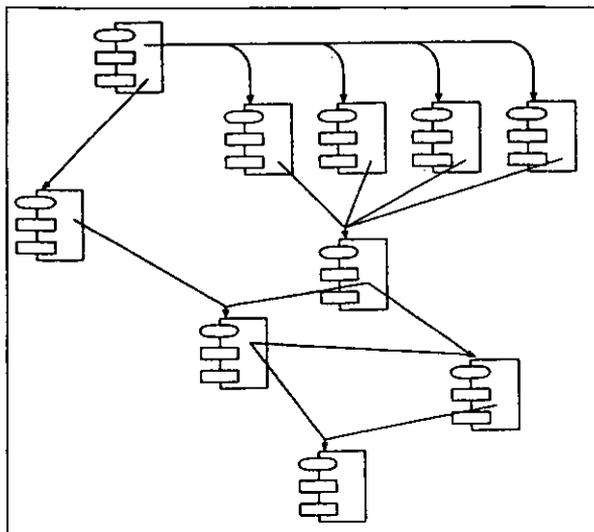


Figura 4.6 Programación orientada a objetos.

4.3 Modelo de Ciclo de Vida y Paradigma de Programación Seleccionado.

El modelo de ciclo de vida a utilizar en la presente tesis se apegará mucho al de entrega por etapas con un paradigma de programación orientado a objetos, teniendo partes funcionales en un tiempo X_1 , X_2 y X_3 , de la siguiente manera:

1. Se tiene en estos momentos una definición clara de lo que el sistema podría hacer.
2. Se tendrá una etapa muy completa de análisis de requerimientos con un experto en el área, definiendo los análisis a efectuar, sus entradas, cálculos y maneras de representar las salidas.
3. Se planteará una aproximación de la manera de trabajar con el software.

4. Se definirá una matriz de selección, la cual nos permitirá ordenar aspectos del software por su importancia (de mayor a menor). Dicha matriz contendrá aspectos técnicos de la implantación (dificultad, tiempo, costo), relevancia para el área del análisis y posición que nos daría en el mercado. De tal manera de poder contar con un documento que nos permita planificar cada una de las tres etapas (los análisis de requerimientos que se llevan hasta ahora sugieren que $X_1=50$ % de funcionalidad, $X_2=70$ % y $X_3=100$ %).
5. Aplicación de la matriz de selección, de la cual se obtendrá una jerarquía en importancia de requerimientos e inclusive se podrían eliminar algunos de los requerimientos especificados.
6. Comenzar con el diseño del sistema con pseudocódigo de los análisis, lo cual nos permitirá definir las estructuras de datos más apropiadas y poder cuidar el desempeño.
7. Diseño de las bases de datos, y los procesos que se encuentran asociados a ellas (Ingreso de la información, normalización y salida de datos).
8. Generación de un prototipo de interfaz (Que muy probablemente sea utilizado en la implementación final) y un manual de usuario, el cual será mostrado a un selecto grupo de investigadores para su evaluación y así tener una retroalimentación que nos permita afinar nuestro producto.
9. Evaluación e incorporación de las modificaciones pertinentes.
10. Se diseñará en un principio en una orientación estructurada que permitirá una mayor facilidad y coherencia de la definición de los objetos.
11. Diseño de objetos: Número, función que desempeñan, comunicación entre ellos e interacción con el usuario.
12. Plan de trabajo de las etapas.
13. Etapa $X_1=50$ % Producto en fase Beta 1 funcional.
14. Etapa $X_2=70$ % Producto en fase Beta 2 funcional.

15. Etapa X₃=100 % Producto en fase Beta 3 funcional.
16. Pruebas finales globales, generación de los manuales finales (en el caso del manual de usuario se deberá tener una muy buena aproximación en la fase del prototipo).

5 Análisis de Requerimientos

Para la realización de esta etapa, se mantuvo una agenda de reuniones con el experto en el área, que permitió convertir la vaga idea del software deseado, a una visión más concreta y delimitada de lo que se quería.

Esta etapa se inicia con el planteamiento que el SIG que se va a desarrollar, es del tipo raster, es decir, los análisis partirán de una rejilla que contendrá información de la presencia/ausencia de una especie. Ninguno de los análisis a efectuar necesita información topológica y la definición de la rejilla puede ser relativamente pequeña.

Aunque se detectó la necesidad de contener y desplegar información cartográfica para referencia de manera vectorial, se acordó que esta característica podría posponerse hasta el desarrollo de la segunda versión, en la cual además se implementaría la manera de importar esta información desde diferentes formatos comerciales (como son DXF/DWG para Autocad, MIF/TAB para MapInfo, DGN para MicroStation, etc.).

En este punto se definió un lenguaje que nos permitirá una mejor comunicación entre el experto y el equipo de desarrollo, por lo cual se llegó a los siguientes conceptos:

1. **Área de Estudio (AE):** Es el área definida por los puntos máximos. Es en sí la región geográfica a dividir con una rejilla determinada para poder aplicar los análisis. Se tendrá una base de datos (BD) por cada AE, que contenga toda la información de ésta.
2. **Unidad Geográfica (UG):** Pequeñas áreas geográficas en la que se divide el AE. Las UGs son cuadriláteros de la misma área y contienen la información de presencia/ausencia de las especies recolectadas.
3. **Rejilla (Rj):** Cuando se define el tamaño de la AE y el tamaño de las UGs. El AE es dividida en tantas UGs como pueda contener (no importando si dicha UG se

considera o no dentro de los análisis). Una BD puede contener varias definiciones de rejillas de una AE.

4. **GeoFrame (GF):** Rejilla en la cual se han seleccionado las UG's que participarán en los cálculos de los análisis. Una definición de rejilla puede tener varias definiciones de Geoframe. Esta funcionalidad del geoframe permite excluir UGs, ya sea porque no tienen datos de recolecta y/o por algún criterio del usuario.
5. **Camino Metodológico:** A algunos análisis se les permitirá el encadenamiento, es decir, los resultados de un análisis X podrán ser parámetros del análisis Y, permitiéndole al usuario el guardar este camino. Un geoframe contiene varios caminos metodológicos.
6. **Localidad:** Se refiere al punto geográfico, de dónde se han obtenido muestras o recolectas de especies. Estos datos son almacenados en una BD, para permitir crear la información de presencia/ausencia de las especies para cada rejilla definida.

Con esta base, se diseñó una metodología para llevar a cabo esta etapa, en la que el primer paso consistió en hacer un listado de los análisis requeridos, los parámetros que requieren, la manera de calcularlos y las salidas que producen. Hecho esto, se pensó en una serie de criterios que nos permitieran jerarquizar dichos análisis, para la planeación de entregas por etapas.

El último paso consistió en obtener los requerimientos de la interfaz, en este punto no se buscó un diseño de ésta, sino una visión muy general de lo que se necesitaba, aunque cabe aclarar, que una de las características que debe poseer el producto final, es su fácil uso, por lo que en la etapa de diseño de la interfaz se debe tener especial cuidado en que sea consistente, coherente y lo más similar a cualquier aplicación del tipo Windows, para facilitar su uso.

5.1 Listado de Análisis Biogeográficos a Realizar.

A continuación se muestran cada uno de los análisis que el software será capaz de efectuar, siendo éste uno de los documentos generados en la fase de análisis de requerimientos.

5.1.1 Diversidad.

Notación :

LSp_i : Lista de especies en la cuadrícula i (*especie 1, especie 2, especie 3, ..., especie n*).

$|LSp_i|$: Número de especies diferentes en la cuadrícula i (cardinalidad). (5)

m : Número de UG's en el Geoframe. (10)

m_i : UG de i .

5.1.1.1 Alfa (α)

Es el número de especies por cada UG.

- **Fórmula :** $\alpha_i = |LSp_i| \forall m_i$ donde $i \in [1, m]$.
- **Entrada :** LSp_i para cada m_i .
- **Salida:** Un entero para UG, número exacto al ser seleccionada la UG.

m número de α 's representados en un mapa temático de n colores por rangos de α y/o mapa con el α_i de m_i .

Ejemplo:

5	4	2	5
1		3	1
	5	1	1

	6-4
	3-2
	1
	N/A

5.1.1.2 Diversidad Alfa Promedio (α):

Es el promedio de todas las diversidades alfa.

• **Fórmula:** $\alpha = \frac{(\sum_{i=1}^m \alpha_i)}{m}$.

- **Entrada:** α_i para $i \in [1,m]$. Un entero para cada cuadro.
- **Salida:** Un entero para toda el AE.

5.1.1.3 Diversidad Gama (γ)

Es el número de especies en total del AE.

Fórmula: $\gamma = \left| \bigcup_{i=1}^m LSp_i \right|$.

- **Entrada:** LSp_i para m_i donde $i \in [1,m]$. Un entero para cada cuadro.
- **Salida:** Un entero para toda el AE.

5.1.1.4 Diversidad Beta:

Es la tasa de cambio de las especies con respecto a las UG's.

5.1.1.4.1 (Wittaker, 1972)

- **Fórmula:** $\beta = \frac{\gamma}{\alpha}$.
- **Entrada:** α y (γ).
- **Salida:** Un real para toda el AE.

5.1.1.4.2 (Wittaker, 1972)

- **Fórmula:** $\beta_w = \frac{k}{\ln 2}$.

Donde k es la constante del ajuste a la curva exponencial del tipo: $y = be^{-kx}$

- **Procedimiento:** Se obtiene la matriz de similitud, dado un criterio ($S_{i,j}$):

	<i>1</i>	<i>i</i>	<i>m</i>		
A1	A2	A3	A4		
<i>1</i>	$S_{1,2}$	$S_{1,3}$	$S_{1,4}$	A1	<i>1</i>
	<i>1</i>	$S_{2,3}$	$S_{2,4}$	A2	<i>j</i>
		<i>1</i>	$S_{3,4}$	A3	
			<i>1</i>	A4	<i>m</i>

Se obtiene la matriz pero ahora de distancias entre todas la UG's (centros geométricos).

	<i>1</i>	<i>i</i>	<i>m</i>		
A1	A2	A3	A4		
<i>0</i>	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	A1	<i>1</i>
	<i>0</i>	$X_{2,3}$	$X_{2,4}$	A2	<i>j</i>
		<i>0</i>	$X_{3,4}$	A3	
			<i>0</i>	A4	<i>m</i>

Y se tabulan de la siguiente forma (sin tener en cuenta la diagonal) :

$m_{i,j}$	$S_{i,j}$	$X_{i,j}$
$m_{1,1}$	$S_{1,1}$	$X_{1,1}$
$m_{1,2}$	$S_{1,2}$	$X_{1,2}$
...
$m_{i-1,j-1}$	$S_{i-1,j-1}$	$X_{i-1,j-1}$

Y la columna $S_{i,j}$ es la variable dependiente de la columna $X_{i,j}$, ajustándose a una curva exponencial a través de una regresión de mínimos cuadrados.

- **Entrada** : Criterio de Similitud a utilizar, matriz de similitud completa, matriz de distancias entre UG's.
- **Salida**: Un real para toda el AE.

5.1.2 Estimadores de Riqueza (S_{est}).

Los estimadores de riqueza son una manera de calcular el número total de especies presentes en AE.

Notación :

$S_{p_{infr}}$: Número de especies no frecuentes (aquellas presentes en 10 ó menos UG).

$S_{p_{frec}}$: Número de especies frecuentes (aquellas presentes en 11 ó más UG).

S_{obsi} = número de especies en m_i .

Q_j : Número de especies que están en j UG's (Q_1 Es el número de especies que se colectaron en una sola UG).

m_{infr} : Número de UG que tienen cuando menos una especie infrecuente.

p_k : Proporción de UG que contienen a la especie k -ésima.

- Salida para todos los estimadores: Un real para toda el AE.

5.1.2.1 ICE :

- **Fórmula:** $S_{est} = S_{frec} + \frac{S_{infr}}{C_{ice}} + \frac{Q_1}{C_{ice}} \gamma_{ice}^2$.

Donde :

$$C_{ice} = 1 - \frac{Q_1}{N_{infr}}$$

$$\gamma_{ice}^2 = \max \left\{ \frac{S_{infr}}{C_{ice}} \frac{m_{infr}}{(m_{infr} - 1)} \frac{\sum_{j=1}^{10} j(j-1)Q_j}{(N_{infr})} - 1, 0 \right\}$$

$$N_{infr} = \sum_{i=1}^{10} (Q_i * i)$$

- **Entrada :** Cálculo de los parámetros.

5.1.2.2 Chao 2:

- **Fórmula :** $S_{est} = S_{obs} + \frac{Q_1^2}{2Q_2}$
- **Entrada :** Cálculo de los parámetros.

5.1.2.3 Jack 1 :

- **Fórmula :** $S_{est} = S_{obs} + Q_1 \frac{m-1}{m}$
- **Entrada :** Cálculo de los parámetros.

5.1.2.4 Jack 2 :

- **Fórmula :** $S_{est} = S_{obs} + Q_1 \left(\frac{2m-3}{m} \right) - Q_2 \left(\frac{(m-2)^2}{m(m-1)} \right)$
- **Entrada :** Cálculo de los parámetros.

5.1.2.5 Boot :

- **Fórmula :** $S_{est} = S_{obs} + \sum_{k=1}^a (1-p_k)^2$
- **Entrada :** Cálculo de los parámetros.

5.1.2.6 MMeans :

- **Fórmula :** $S(m) = \frac{S_{est} m}{B + m}$
- **Entrada :** π_i para cada m_i donde $i \in [1, m]$ y el número de iteraciones.

m_1	π_1	$\pi_1 = LSp_1 $
M_2	π_2	$\pi_2 = LSp_1 \cup LSp_2 $
....	...	
m_i	π_i	$\pi_i = LSp_1 \cup LSp_2 \cup \dots \cup LSp_i $

- **Procedimiento** : Transformar los pares (m_i, π_i) en $(\frac{1}{m_i}, \frac{1}{\pi_i})$; realizar ajuste de mínimos cuadrados donde :

$$Y = \text{Pendiente} \left(\frac{1}{\pi} \right) + \text{Ordenada_origen}$$

$$S_{est} = \sum_{i=1}^{n_int\ eracion} \frac{1}{\frac{ordenada_origen_i}{n_int\ eracion}}$$

5.1.3 Índices de Similitud.

Permiten obtener mapas de similitud de UG's, además de ser utilizados en los agrupamientos.

Notación :

S_i : *Similitud*. Similitud entre un par de UGs.

$LSp_i \cap LSp_j$: *Intersección*. Elementos en común en Sp_i y Sp_j .

$LSp_i \cup LSp_j$: *Unión*. Elementos que se encuentran en Sp_i o en Sp_j .

$|LSp_i|$: *Cardinalidad*. Número de elementos en un conjunto.

$\overline{LSp_i}$: *Complemento*. Elementos del universo que no están en Sp_i .

$\max(|LSp_i|, |LSp_j|)$: *Función*: máximo de dos valores enteros.

$\min(|LSp_i|, |LSp_j|)$: *Función*: mínimo de dos valores enteros.

Donde i y $j \in [1, m]$

- **Entrada :** Sp_i para todo m_i donde $i \in [1,m]$. Para observar un mapa de similitud con respecto a una UG hay que seleccionar la UG pivote (un renglón de la matriz de similitud) .
- **Salida:** Puede ser un mapa temático de n colores que represente un renglón de la matriz, dado un criterio.

Toda la matriz en forma de reporte y la matriz coloreada.

Mapa Temático:



	100%-90%
	80%-60%
	<50%
	N/A
	UG. Pivote

Matriz Reporte:

A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	
100	0	0	0	0	0	0	0	0	0	0	0	A1
	100	0	0	0	50	50	0	100	50	33	0	A2
		100	50	0	50	0	50	0	0	33	0	A3
			100	0	33	33	100	0	33	25	0	A4
				100	0	0	0	0	0	0	0	B1
					100	33	33	50	33	60	0	B2
						100	33	50	100	25	0	B3
							100	0	33	25	0	B4
								100	50	33	0	C1
									100	33	0	C2
										100	33	C3
											100	C4

Matriz Coloreada:

A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4	
												A1
					50	50			50	33		A2
			50		50		50			33		A3
					33	33			33	25		A4
												B1
						33	33	50	33	60		B2
							33	50		25		B3
									33	25		B4
									50	33		C1
										33		C2
											33	C3
												C4

5.1.3.1 Braun-Blanquet :

- Fórmula : $S_i = \frac{|LSp_i \cap LSp_j|}{\max(|LSp_i|, |LSp_j|)}$

5.1.3.2 Fager :

- Fórmula: $S_i = \frac{|LSp_i \cap LSp_j|}{\sqrt{|LSp_i| |LSp_j|} + \frac{\max(|LSp_i|, |LSp_j|)}{2}}$

5.1.3.3 Jaccard :

- Fórmula: $S_i = \frac{|LSp_i \cap LSp_j|}{|LSp_i \cup LSp_j|}$

5.1.3.4 Kulezynski 1:

- Fórmula : $S_i = \frac{|LSp_i \cap LSp_j|}{|LSp_i \cup LSp_j| - |LSp_i \cap LSp_j|}$

5.1.3.5 *Kulezynski 2*

- **Fórmula:** $S_i = \frac{|LSp_i \cap LSp_j| |LSp_i \cup LSp_j|}{2|LSp_i| |LSp_j|}$

5.1.3.6 *Otsuka :*

- **Fórmula:** $S_i = \frac{|LSp_i \cap LSp_j|}{\sqrt{|LSp_i| |LSp_j|}}$

5.1.3.7 *Simpson:*

- **Fórmula:** $S_i = \frac{|LSp_i \cap LSp_j|}{\min(|LSp_i|, |LSp_j|)}$

5.1.3.8 *Srensen-Dice :*

- **Fórmula:** $S_i = \frac{2|LSp_i \cap LSp_j|}{|LSp_i| + |LSp_j|}$

5.1.3.9 *Radio de correlación:*

- **Fórmula:** $S_i = \frac{|LSp_i \cap LSp_j|}{\sqrt{|LSp_i| |LSp_j|}}$

5.1.3.10 *Baroni-Urbani-Buser :*

- **Fórmula:** $S_i = \frac{|LSp_i \cap LSp_j| - \sqrt{|LSp_i \cap LSp_j| |LSp_i \cup LSp_j|}}{|LSp_i \cap LSp_j| + \sqrt{|LSp_i \cap LSp_j| |LSp_i \cup LSp_j|}}$

5.1.4 Calidad de Muestreo (Q_s).

• **Fórmula :**
$$Q_s = \frac{I}{(S_{obs})^m / (1 - E_s) - \max(S_{obs}, m)}$$

Donde :

$$E_s = \frac{S_{est} - S_{obs}}{S_{est}}$$

$$I = \sum_{j=1}^m \alpha_j$$

$q(j)$ es el número de especies que están en j áreas (I es el número de 1s en la matriz de presencia/ausencia).

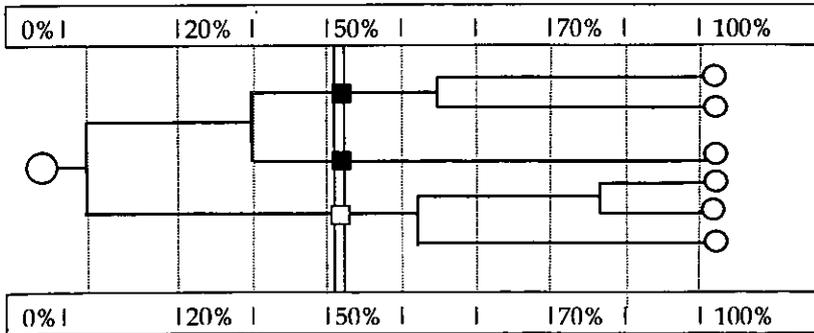
- **Entrada :** Cálculo de los parámetros, alfa, seleccionar índice de estimación.
- **Salida :** Un número real para toda la AE.

5.1.5 Ligamientos.

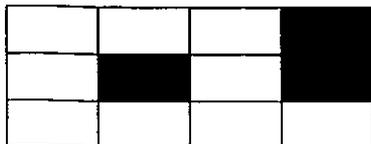
- **Entrada :** Índice de similitud, resultados de éste y nivel de corte (opcional).
- **Salida :** Un fenograma en caso de no especificar nivel de corte.

Un mapa temático de agrupamiento de UG dado un nivel de corte.

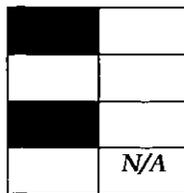
Fenograma:



Mapa Temático:



Nivel de corte =47%



- **Ligamiento promedio:** Se calcula el promedio de la similitud de los dos agrupados.
- **Ligamiento completo:** Se calcula el mayor de la similitud de los dos agrupados.
- **Ligamiento simple:** Se calcula el menor de la similitud de los dos agrupados.

5.1.6 Elección de Áreas para la Conservación:

(Método Iterativo, Rebelo, 1994)

- **Procedimiento:**

Consiste en elegir una lista de UG que contenga a todas las especies de la región.

Supóngase la siguiente matriz de presencia/ausencia (las letras son las UG y los números las especies):

<i>m_i/Sp_i</i>	A	B	C	D	E
1	X	X	X	X	
2		X	X	X	
3			X		X
4				X	X

Se procede a calcular un número de importancia de cada especie a partir del número de UG en que se encuentra:

Importancia

m_i/Sp_i	A	B	C	D	E	Suma	$(5/Suma)^1$
1	X	X	X	X		4	1.25
2		X	X	X		3	1.66
3			X		X	2	2.5
4				X	X	2	2.5

Se le asigna una importancia a cada UG, como una suma las importancias de especies que contiene :

Iteración 1

A	B	C	D	E	Importancia del UG.
1.25	2.91	5.41	5.41	5	$\Sigma(5/Suma)$
		4			Cuadro elegido para la reserva

En la siguiente iteración sólo se suman las importancias de las especies que no están en un UG elegido como reserva (no se suman las especies 1, 2 y 3), en este caso sólo cuenta la especie 4, que tiene un valor de 2.

Importancia

m_i/Sp_i	A	B	C	D	E	Suma	$(4/Suma)$
1						0	0
2						0	0
3						0	0
4				X	X	2	2

Iteración 2

A	B	C	D	E	Importancia del UG.
		4	2	2	$\Sigma(4/Suma)$
			4		Cuadro elegido para la reserva

Se propone conservar a dos UG (C, D) pues en estas dos UGs se captura toda la diversidad de la región.

- **Entradas:** Datos solicitados y el número de la iteración de paro (opcional)
- **Salida:** Mapa Temático con un orden cardinal.

Mapa Temático:

1	2
2	2
	2

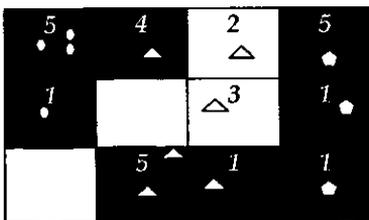
	1
	2
	N/A

5.1.7 Localidades de Recolecta.

- **Entradas :** Localidades contenidas en UG's.
- **Salidas :** Mapa con marcadores clasificados (por forma) según los siguientes criterios:
 - Número de especies.
 - Número de visitas.
 - Número de colectores.
- **Comentarios :** Los puntos no son sensibles al click. Se puede ver el mapa de diversidad alfa, en estos momentos las UG's son dinámicas, pero una vez definida la rejilla ya no lo son.

Ej.:

Mapa



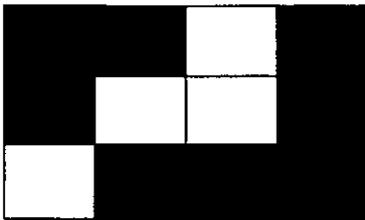
Alfa NUM de visitas

	6-5	o	5
	3-2	△	4 a 3
	1	◻	2 a 1
	N/A		

5.1.8 Especies → Áreas

¿Dónde está un determinado conjunto de especies? (Área de distribución), ¿dónde la cardinalidad del conjunto n es de $[1, \gamma]$?

- **Entradas** : Seleccionar 1 ó más especies, criterio (And =Presencia/ausencia, Or=Grados de Presencia).
- **Salidas** : Mapa temático de 3 colores (presencia ausencia y N/A) o mapa temático de n colores (grado de presencia)



	100%	<i>Lista de Especies:</i> <i>Especie 1</i> <i>Especie 2</i> <i>Especie 3</i> <i>Especie n</i>
	99-1%	
	0%	
	N/A	

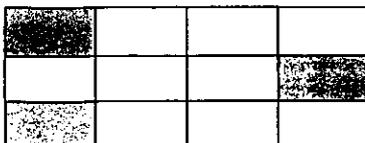
5.1.9 Áreas → Especies

5.1.9.1 En un conjunto de UG

¿Qué especies hay en conjunto de UG's (OR y AND) donde la cardinalidad del conjunto n es de $[1, m]$?

- **Entradas** : Seleccionar el conjunto de UG's.
- **Salidas** : $LSp_{[1, n]}$, Lista de especies de un intervalo de UG's (Unión o intersección)

Ej. :



<i>Lista de Especies:</i>	
<i>Especie 1</i>	
<i>Especie 2</i>	
<i>Especie 3</i>	
<i>Especie n</i>	
<i>Operador : AND</i>	

5.1.9.2 De toda el área.

- **Entrada:** LSp_i donde $i \in [1,m]$.
- **Salida:** $\bigcup_{i=1}^m LSp_i$ donde $\left| \bigcup_{i=1}^m LSp_i \right| = \gamma$.

Ej. :

<i>Lista de Especies:</i>
<i>Especie 1</i>
<i>Especie 2</i>
<i>Especie 3</i>
...
<i>Especie γ</i>

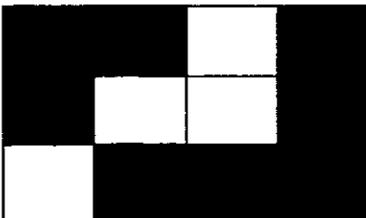
5.1.10 UG's.

¿Cuáles UG contienen n especies, donde $n \in [1,gama]$?

- **Entrada :** P_n, n .
- **Salida:** Mapa temático de 3 colores (si/no/NA).

Ej. :

$P_n=2$ (UG's con solo dos especies).



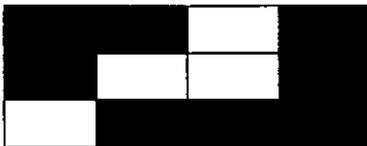
	<i>SÍ</i>
	<i>NO</i>
	<i>N/A</i>

5.1.11 Especies

¿Cuáles especies están en n UG's (especies de distribución restringida)? ¿Cuáles especies están en dos cuadros, tres,...,m?.

- Entradas : Q_n , n.
- Salida :
 - Lista de especies Q_n .
 - Mapa temático de Q_n en función de cuántas especies Q_n contiene.

Ej. :



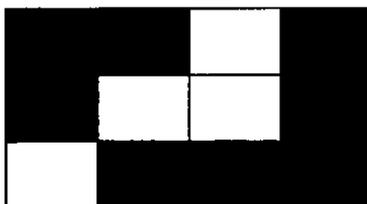
	7 a 5	Lista de Especies Q_2 : Especie 1 Especie 2 Especie 3 ... Especie n
	4 a 1	
	0	
	N/A	

5.1.12 Entre UG's.

¿Cuánto se parecen dos conjuntos de UG's: donde la comparación puede ser 1 a 1, 2 a 5, ..., A a complemento de A, etc.?

- Entrada : selección de los dos conjuntos de UG's, Selección de criterio de similitud.
- Salida : Número real y una lista de especies comunes y totales (unión intersección).

Ej. :

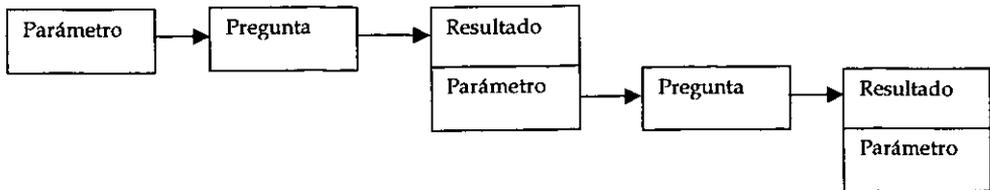


Intersección. Unión.	Lista de especies Especie 1 Especie 2 Especie 3 ... Especie n

Similitud entre los ■ y ■ = 75%. Criterio : Fager.

5.2 Comentarios del Listado de Análisis.

En general se permitirá la transitividad siempre que sea posible.



Para este fin se define una gramática como, donde cada análisis será un símbolo y la regla de construcción será una matriz que indique el símbolo que le puede seguir. La gramática así definida es:

Símbolo.	Descripción	Entrada	Salida
Alfa	Mapa Alfa	UG	UG
Sim	Renglón de la Matriz de Similitud	UG	UG
Liga	Ligamiento	UG	UG
Rebelo	Elección de áreas para la conservación	UG	UG
SpUg	Donde esta determinado conjunto de especies (Especies → Areas)	SP	UG
UgSP	Qué especies hay en un conjunto de UGs (Areas → Especies)	UG	SP
TotUgSp	Lista de especies de toda el área	UG	SP
UgnSpUg	Cuáles UGs contienen n especies (UGs)	UG,n	UG
SpnUgSp	Cuáles especies están en n Ugs (Especies)	SP,n	SP
UgvsUgSp	Cuánto se parecen dos conjuntos de Ugs	UG,UG	SP

Agrupando por el tipo de entrada→salida obtenemos que:

Mapa De →A	Total	Lista de Símbolos.
UG→UG	5	Alfa, Sim, Liga, Rebelo, UgnSpUg.
UG→SP	3	UgSp, TotSp, UgvsUGSp.
SP→UG	1	SpUg
SP→Sp	1	SpnUGSp

Por lo que la matriz de construcción es la que sigue:

Entrada

		Alfa	Sim	Liga	Rebelo	SpUg	UgSp	TotUgSp	UgnSpUg	SpnUgSp	UgvsUgSp	
		GU	GU	GU	GU	GU	SP	SP	GU	SP	SP	
Salida	Alfa	GU	S									
	Sim	GU		S								
	Liga	GU			S							
	Rebelo	GU				S						
	SpUg	SP					NA					
	UgSp	GU						NA				
	TotUgSp	NA							NA			
	UgnSpUg	GU								NA		
	SpnUgSp	SP									NA	
	UgvsUgSp	UG										NA

	Es Posible
NA	No es posible la recursividad
	No es posible
S	Recursividad Posible

La recursividad significa:

	Alfa	Sim	Liga	Rebelo	SpUg	UgSp	TotUgSp	UgnSpUg	SpnUgSp	UgvsUgSp
Alfa	Re-jerarquización (cuando las clases están en automático)									
Sim	Navegación por la Mat.Sim.; Refinamiento de las clases									
Liga	Refinamiento; cambio en el nivel de corte. El árbol no cambia									
Rebelo	Se podría elegir las 2-n prioritarias y reanalizar									

5.3 Matriz de Decisión.

La matriz de selección nos permitirá, una vez terminado el análisis de requerimientos, el poder clasificar los análisis en función de criterios objetivos.

Esta clasificación será de gran ayuda para establecer el plan de desarrollo, en donde, de ser posible, las partes más importantes se desarrollen primero.

Los criterios se evalúan de 1 a 10, donde 1 es lo menos y 10 el más, es decir se clasificarán los requerimientos de mayor a menor puntuación y se refieren a todos los conceptos asociados con el área a la que apoyará este software, así como criterios que maximicen su mercado potencial:

- Estándar/Innovador (ponderación 1): Éste es en realidad un criterio doble del tipo O exclusivo, en caso de que el análisis sea estándar se le verá una importancia con respecto a los estándares y de manera similar en los innovadores, se le dará una calificación de 10 al más atractivo.
- Porcentaje de usuarios (ponderación 2): Se refiere al número de usuarios que van a usar el software por haber incorporado X característica. Donde 1 será el 10% de los usuarios y 10 el 100% de ellos.
- Complejidad del concepto (ponderación 2): Se refiere a qué tan bien comprendido es el concepto a implementar. 1 es lo difícil de entender y 10 lo más fácil de entender.
- Está ya implementado (ponderación 1): Se refiere a si el análisis en cuestión se encuentra ya implementado en el mercado. Donde 1 ya se encuentra y 10 no se encuentra.

Una vez definidos los criterios, las pláticas con el experto en el área arrojaron los siguientes resultados:

	Análisis	Innovación/Estándar	r. Atractivo	r. vulnerable	r. usuarios. Lo van a usar	Complejidad de los conceptos	Ya está implementado	TOTAL	60-
	Peso	1	2	2	2	1	6		
Diversidad	Div. Alfa	10	10	10	8	58	2		
	Div. Alfa promedio	9	9	8	9	52	8		
	Div. Beta (G/A)	10	6	5	10	42	18		
	Div. Beta (exponencial)	10	6	3	10	38	22		
	Div. Gama	8	10	10	8	56	4		
Estim. de Riqueza	ICE	10	8	6	8	46	14		
	Chao 2	9	8	6	8	45	15		
	Jack 1	8	8	6	8	44	16		
	Jack 2	8	8	6	8	44	16		
	Boot	8	8	6	8	44	16		
	Mmenten	8	8	6	8	44	16		
B. Cuant.	Ind. similitud	10	8	7	8	48	12		
	Qs	6	6	5	10	38	22		
	Ligamientos	8	10	10	8	56	4		
	Rebelo	9	7	8	10	49	11		
Preguntas	Núm. de especies	10	8	10	9	55	5		
	Núm. de visitas	8	5	9	10	46	14		
	Núm. de colectores	6	5	8	10	42	18		
	Área de distribución	10	10	10	10	60	0		
	Lista de especies	10	10	10	10	60	0		
	UGs por riqueza	6	5	5	10	36	24		
	Especies por distribución	8	6	6	10	42	18		
	Similitud entre UGs	7	7	8	10	47	13		

Ordenando por la columna 60- se obtiene el orden de importancia:

Importancia	Análisis
0	Área de distribución
0	Lista de especies
2	Div. Alfa
4	Div. Gama
4	Ligamientos
5	Núm. de especies
8	Div. Alfa promedio
11	Rebelo
12	Ind. similitud
13	Similitud entre UGs
14	ICE
14	Núm. de visitas
15	Chao 2
16	Jack 1
16	Jack 2
16	Boot
16	MMMean
18	Div. Beta (G/A)
18	Núm. de colectores
18	Especies por distribución
22	Div. Beta (exponencial)
22	Qs
24	UGs por riqueza

5.4 Requerimientos de la Interfaz.

En esta etapa del análisis se pensó en la forma de presentar los resultados, ingresar los parámetros para obtener los resultados, y una serie de operaciones que permitieran el manejo del sistema resultante.

La 1era parte que se analizó en los requerimientos de la interfaz, fue los resultados de tipo mapa y fenograma, indicando qué tipo de operaciones y características debían tener, la 2da parte fue poner en concreto el tipo de pantallas de salida que tendría el sistema, y la tercera mucho más formal y específica en la cual se

agrupan y clasifican los análisis anteriormente descritos y las demás operaciones que debía tener el sistema en un conjunto de menús.

5.4.1 Mapa :

En esta parte se especificó un conjunto de operaciones gráficas sobre el mapa, así como su comportamiento y contenido como se describe a continuación:

A) Operaciones gráficas del mapa:

- Acercar/Alejar (Zoom).
- Movimiento dentro del mapa (Pan).
- Cada UG Sensible al botón izquierdo para seleccionarla.
- Todo el mapa sensible al botón derecho para menús contextuales.
- Sensible a la posición del ratón para el despliegue de coordenadas y análisis por cada UG.

B) Se debe poder imprimir su contenido.

C) Se podrá seleccionar entre una paleta de colores predeterminada para la generación de los mapas temáticos (esquema semáforo, grados de verde, contrastante, etc.).

D) Se debe poder visualizar la regla.

E) Debe aparecer algún elemento gráfico que denote la escala.

F) Debe aparecer algún elemento gráfico que denote el norte.

G) Se debe permitir el encendido/apagado de la retícula.

H) Se deben manejar todos los cuadrantes de la Tierra (despliegue UTM.).

I) Se podrán desplegar algunos elementos de cartografía base (**SEGUNDA VERSIÓN DEL SW**):

a) **Texto:** aparecerá texto en la pantalla con una determinada orientación y tamaño de letra.

b) **Puntos y líneas:** se permitirán trazado de puntos y líneas, éstas últimas con calidad de línea y color.

c) **Polígono:** se permitirá al seleccionar las UG's y el relleno de éstos.

J) **Importación de archivos DXF (SEGUNDA VERSIÓN DEL SW):**.

5.4.2 Matriz coloreada:

Se definieron las operaciones a realizar en la matriz coloreada:

- A) Operaciones acercarse y alejarse (Zoom).
- B) Desplazamiento (Pan).
- C) Simbología, paleta de colores predeterminada y seleccionable.

5.4.3 Fenograma:

Las operaciones del fenograma son las que siguen:

- A) Operaciones de acercarse y alejarse (Zoom).
- B) Desplazamiento (Pan).
- C) Regla visible.
- D) Mover el nivel de corte dinámicamente.

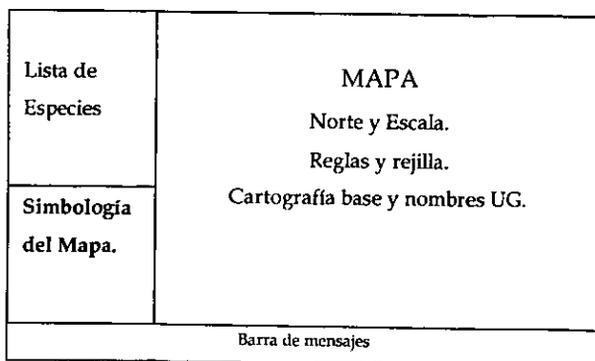
5.4.4 Pantallas de Salida:

Se definieron 3 tipos de pantalla de salida o de presentación de los resultados, las cuales serán el documento de trabajo y tendrán el comportamiento especificado anteriormente. Las pantallas de salida se esquematizan de la manera siguiente:

- Pantalla Salida Tipo I.

Subcasos :

1. Maximizar mapa.
2. Maximizar mapa y simbología.
3. Maximizar lista especies.



Pantalla de Salida Tipo II.

Subcasos:

1. Maximizar mapa.
2. Maximizar mapa y simbología.
3. Maximizar fenograma.

Fenograma Reglas y barras de desplazamiento.	
Simbología del Mapa.	MAPA Norte y Escala. Reglas y rejilla. Cartografía base y nombres UG.
Barra de mensajes	

5.4.5 Menús:

Una vez que se tiene claro cada uno de los análisis: el tipo de entradas y el tipo de salidas, las demás operaciones que se necesitan para poder trabajar con el sistema creado, en esta última parte del análisis de requerimientos, se especificó la estructura de menús que agrupa de manera ordenada todas las operaciones.

Los menús propuestos son:

1. Archivo
2. Edición
3. Ver
4. Análisis
5. Preguntas
6. Herramientas
7. Ventana
8. Ayuda

La descripción menú por menú es la que sigue:

1) Archivo	2) Edición	3) Ver
Nueva AE...	Nombre de la UG.	Acercar
Abrir AE...	Copiar	Alejar
Abrir Geoframe...		Ver retícula
Importar Datos		Ver reglas
Exportar Datos		Desplazamiento
Guardar		Ver toda el AE.
Guardar como..		Información de la UG
Imprimir		
Salir		

4) Análisis		5) Preguntas	
Mapa Alfa		Localidades	
Mapa de Similitud		Áreas de Distribución	
Matriz de Similitud->	Matriz coloreada	Lista de Especies ->	De toda el AE.
Bioestadísticas	Matriz %	UG's por Riqueza	UGS Seleccionadas
Agrupamiento		Especies por Distribución	
Áreas Prioritarias		Similitud entre UG's	
		Limpiar.	

6) Herramientas		7) Ventana
Opciones		Subcasos de ventanas
Paletas y Rangos		
Definir Rejilla		
Seleccionar UG ->	Selección manual	
Definir Geoframe	Selección por rangos	
Camino Metodológico	Seleccionar toda el AE	
Usuarios		

En la siguiente etapa, el diseño, se especificará de manera más clara el comportamiento de cada uno de estos menús y de las demás partes del sistema, que aunque el usuario no las perciba directamente, son de vital importancia (ejm. la base de datos).

6 Diseño

Una vez concluida la etapa de análisis de requerimientos, se procedió a realizar el diseño del software. Esta etapa consiste en aplicar una serie de reglas que nos conduzcan a la solución del problema proporcionando un camino para convertir los requerimientos en un sistema. El propósito del diseño en el desarrollo de SI es el de construir un sistema que:

- Satisfaga un conjunto de especificaciones funcionales.
- Que se ajuste a las limitaciones del medio.
- Conozca explícita o implícitamente los requerimientos en cuanto al desempeño y uso de recursos.
- Satisfaga explícita o implícitamente un criterio de diseño en cuanto a la forma.
- Satisfaga requerimientos propios del diseño, tales como el tamaño, costo o las herramientas disponibles para el desarrollo (Mostow, 1985).

Esta etapa debe concluir con el diseño total del sistema, no dejando ninguna cuestión por resolver en la etapa de implementación, por lo que se debe ejecutar con especial cuidado, pues el omitir algún requerimiento, o el no delimitarlo de manera correcta, puede provocar que el proyecto se salga de tiempo, se incremente su costo, sea inestable, etc.

Para este desarrollo se empezó por modelar la BD que albergaría la información de recolecta, la información necesaria para la creación y manejo del geoframe, así como de almacenar algunos resultados necesarios.

El siguiente paso consistió en formalizar la interfaz, y crear un prototipo con una presentación de la manera de trabajar con el SW al usuario final, para poder contar con una retroalimentación, y así comprobar la consistencia y el fácil uso del desarrollo. Las sugerencias y modificaciones que se generen en el paso anterior, serán evaluadas, para decidir si son incorporadas en el diseño y la manera de hacerlo.

La parte final del diseño consistió en el modelo orientado a objetos que controlaran tanto la interfaz, así como las funciones necesarias para la incorporación de lecturas a la BD, lectura de ésta, realización de los análisis y despliegue de los resultados.

Los pasos anteriores nos darán como resultado una descripción muy precisa de cómo va a estar formado el SI, su comportamiento y su cara ante el usuario. Esta etapa culmina precisamente con 3 documentos con los que se empezará la implementación:

1. La Base de Datos, con el modelo entidad relación y el diccionario de datos (apéndice 2).
2. La Interfaz, con el diagrama de navegación y la funcionalidad de cada diálogo y control (apéndice 3).
3. La jerarquía, comunicación y funcionalidad de las clases (Mostrada más adelante en este capítulo).

Además de una serie de reglas y técnicas de programación necesarias para garantizar la calidad y el fácil mantenimiento del código (apéndice 1), tales como:

1. Notación de las tablas, campos y vistas que conforman la BD.
2. Notación de programación, para el nombramiento de variables, controles, clases, objetos, etc.
3. Reglas para el manejo de errores de ejecución.
4. Reglas de documentación del código.

Teniendo, por último, solucionado todos los problemas técnicos como pudieran ser:

- La plataforma de desarrollo.
- El manejo de los elementos gráficos no aportados directamente por la plataforma de desarrollo, como lo son: el mapa, el fenograma y la matriz.
- La importación de datos.
- El control de los cálculos (desbordamiento de pilas, ciclos infinitos, etc).

6.1 Diseño de la Base de Datos.

Lo primero que se hizo fue revisar otros diseños de BD similares, sobre todo para los datos de recolección. La conclusión de este paso dio como resultado el diseño y creación de una BD prototipo, la cual se deberá probar con un ejemplo real de datos para comprobar su buen diseño.

El primer paso del diseño de la BD consistió en identificar los grandes bloques que la conformaban, posteriormente se identificaron las tablas necesarias y sus relaciones (modelo Entidad Relación: ER) y a través de qué campos. Una vez terminado el modelo ER se agregaron todos los atributos o campos de cada tabla, con su tipo (numérico, texto, etc.) y su longitud (entero, flotante, n caracteres, etc.). Cabe aclarar que antes de proponer el modelo ER, ya se contaba con un documento que definía toda la notación de la BD.

La BD se dividió conceptualmente en 5 grandes áreas:

6.1.1 Los Datos de Recolecta:

Esta parte de la base de datos deberá contener los datos de recolecta (especies, localidades, fechas, etc.) para poder generar tantas rejillas como fueran necesarias para el investigador. El diseño de esta área se apoyó en algunos diseños anteriores para este fin de algunas instituciones nacionales y extranjeras:

- La propuesta por el "Instructivo para la Conformación de Bases de Datos Compatibles con el Sistema Nacional de Información sobre Biodiversidad (SNIB)" de la CONABIO.
- Diseño de referencias del Museo de Paleontología de la Universidad de Berkley, California.
- "A Common Data Estructure for European Floristic Databases (CDEFD) " del BGBM (The Botanic Garden and Botanical Museum Berlin-Dahlem) fundado por la comisión Europea en su programa de Biotecnología. Julio 1996.
- Expert Center for Taxonomic Identification (ETI).

- “The Biodiversity Data Base Manager (BIOTA)” de la Universidad de Connecticut.

Se concluyó que esta parte de la BD debía de ser lo más simple posible, pues no es la finalidad de nuestra herramienta ser un catálogo de riqueza, sino pretende estar en un nivel más alto, haciendo análisis precisamente de estos datos.

Fue aquí, donde se detectó la importancia de los procesos de importación de datos, concluyendo lo siguiente:

1. El diseño de la BD debe ser abierto, es decir la documentación final contendrá el modelo completo, así como el diccionario de datos para que a los usuarios les sea posible modificar el diseño agregando campos e inclusive tablas (apéndice 2).
2. La BD a utilizar en la primera versión será de MS Access, debido a su popularidad, y se planea la conexión a otras BD a través de ODBC (Open Data Base Connect) en la segunda versión.
3. Los procesos de importación serán:
 - 3.1. Importación de nombres de especies de otra BD del mismo programa.
 - 3.2. Importación de datos de recolección por medio de un archivo de texto separado por comas (*.csv) único.
4. Una revisión de integridad opcional, que evitará, en medida de lo posible, información duplicada de datos de recolección (nombre de especies, localidades, etc.) por medio de una revisión de similitud de cadenas.
5. La interfaz de importación permitirá hacer el mapeo de campos desde el archivo de texto hacia los campos que contendrán la información, permitiendo la concatenación de campos cuando sea necesario.

6.1.2 Área del Geoframe:

Esta parte de la BD contendrá todas las tablas que conformen la rejilla y el geoframe:

1. Datos para su creación:
 - 1.1. Puntos más al SW y NE de el AE, de cada uno de las UGs, centroides, etc.
2. Relación de cada UG con las especies que contiene.
3. Resultados de los análisis que se apliquen a todo el geoframe, para incrementar el desempeño del programa.
4. Matriz de similitud para cada criterio.
5. Su vigencia, pues para nuevas recolectas será necesaria la definición de una nueva rejilla (colectas fuera del AE actual) o geoframe (colectas dentro del AE actual) , para mantener su consistencia con los caminos metodológicos.

6.1.3 Caminos Metodológicos:

Esta región de la BD contendrá toda la información requerida para recuperar cualquier paso del encadenamiento de análisis que efectúe el usuario, permitiéndole continuar a partir del último.

Se encontrarán almacenados en esta región:

1. El nombre y orden de los análisis aplicados.
2. Sus parámetros y resultados.
3. Los árboles de similitud requeridos.
4. La información de los rangos utilizados.

6.1.4 Tablas de Rangos y Colores :

En esta parte de la BD estará contenida la información de los rangos y paletas de colores que se pueden usar dentro de la aplicación, así como almacenar los diseñados por el usuario :

1. La información de los rangos:
 - 1.1.El análisis que lo requiere.
 - 1.2.El tipo de rango (manual, percentiles e intervalos constantes).
 - 1.3.El número de intervalos.

- 1.4. Los límites de cada uno de los rangos.
2. La información de las paletas de colores: El programa proveerá de varias paletas de colores predeterminadas, permitiendo al usuario la selección de cualquiera de ellas, ahorrándole así tiempo.
 - 2.1. La definición del color (RGB).
 - 2.2. El número de intervalos para el que está definido.
 - 2.3. Su orden dentro del rango.

6.1.5 Tablas del Sistema:

Esta parte de la BD almacena información varia, como es:

1. Tablas de las plantillas de importación: Permiten almacenar configuraciones sobre el mapeo de campos en la importación de datos hacia el sistema.
2. Información de los usuarios, que de manera opcional le permitirán cierto control sobre los datos.

6.2 Diseño de la Interfaz

En realidad este paso inicia desde la etapa de análisis de requerimientos⁴, y en esta etapa lo que se hace es formalizarla en varios documentos (Apéndice 3) los cuales contienen:

1. La estructura de la barra de menús (Se tenía desde la fase anterior).
2. Los comandos del menú que tienen su representación en la barra de herramientas.
3. La acción de cada uno de los comandos del menú.
4. Cada cuadro de diálogo utilizado, los controles que contiene, su comportamiento, el comando del menú que lo dispara, etc.
5. Los tipos de pantalla de salida (tipo de resultados).

⁴ Una parte del análisis se hizo con la idea de "Sabrás cuando lo veas" y "una imagen vale más que mil palabras".

Cada uno de los puntos anteriores tuvo una evolución, partiendo de una descripción general de la interfaz, hasta convertirse en documentos muy detallados al contener todos los elementos técnicos del diseño, como pudieran ser:

1. Nombres de las formas y los controles que se deberán tener en el código, obedeciendo la reglas de la notación propuesta.
2. Los nombres de los archivos de código fuente.
3. Los identificadores de recursos gráficos utilizados, así como los identificadores de todos los mensajes presentados por el programa.
4. Sus fuentes de datos.
5. Una breve descripción de su funcionalidad y el objeto encargado de su manejo.

Además de los documentos generados en esta etapa, se generó un prototipo de la interfaz, para que fuera mostrado a un selecto grupo de usuarios, con la finalidad de tener una retroalimentación y comprobar ciertos aspectos como lo son:

1. Facilidad de uso: Esta es una de las características más importantes que debe poseer el desarrollo para su comercialización.
2. Coherencia: Se refiere a la idea de que al trabajar con el SW, el usuario no sienta que la interfaz cambia bruscamente de una área a otra del programa (ej. La localización de los botones de aceptar y cancelar dentro de cualquier diálogo).
3. Maximizar la similitud con cualquier aplicación del ambiente Windows: Con la finalidad de reducir al máximo la curva de aprendizaje, al permitir enfocarse a la funcionalidad del software, y no a la manera de trabajar con él.
4. Distribución de los diálogos y pantallas: Que sean atractivas visualmente, equilibradas en su carga de controles, selección de los textos más concisos y claros para los mensajes del programa.

Esta presentación del prototipo se hizo a través de una presentación al usuario final, en donde por medio de una sesión hipotética con el SW se ilustra la manera

de trabajar con el paquete, desde la importación de datos hasta un camino metodológico.

6.3 Modelo de Objetos.

La base de datos, fue un diseño de arriba hacia abajo (top-down), en el caso del diseño de los objetos, se optó por un diseño de abajo hacia arriba (bottom-up), lo cual, debido a la complejidad inherente de cualquier Software, es una práctica muy común usar las dos maneras en el diseño de programas.

El diseño de objetos puede ser un proceso muy difuso, generalmente se recomienda un proceso iterativo e incremental, en el cual, entre más conocimiento tengamos sobre el problema a resolver, más fácil será su solución en cada paso.

El primer paso que se siguió fue el encontrar las clases y sus interrelaciones más fundamentales, se partió de la clave que se desea modelar algún aspecto de la "realidad" (Stroustrup,1993), empezando de abajo hacia arriba, se encontró que los objetos básicos debían ser las especies y las UGs. Las pláticas con el experto durante el análisis y el diseño del sistema fueron cruciales para esta etapa, pues permitió además de la definición de las clases definir las interrelaciones naturales de éstas a través de un proceso que fue de lo más informal a lo más formal: desde una lista de cosas, que podían convertirse en clases o atributos de éstas, hasta la lista y relaciones de las clases encontradas.

El siguiente paso consistió en refinar las clases especificando los conjuntos de operaciones que pueden realizar cada una de ellas a través de la siguiente estrategia :

1. Se consideró cómo se debe construir, copiar y destruir un objeto de la clase.
2. Se definió un conjunto mínimo de operaciones que requiere el concepto representado por la clase.
3. Se consideró qué operaciones se pueden añadir para hacer más fácil su manipulación, incluyendo únicamente las que sean verdaderamente importantes.

4. Se definieron los nombres y funciones que son comunes a todas las clases.

Es muy importante mantener la idea de lo mínimo, pues es más fácil agregar una función a una interfaz, una vez establecida claramente su necesidad, que eliminarla cuando es un estorbo, aunque esto implique más trabajo por parte del diseñador.

Posteriormente se definieron las dependencias de las clases, lo cual es una extensión del paso anterior, aquí se establece exactamente cómo las clases interactúan entre sí en el sistema. Únicamente la relación que se utilizó en este diseño fue el empleo, debido a que no fue necesaria ninguna relación de herencia. Dentro de las relaciones de empleo de las clase se utilizaron funciones que modificaban el estado de algunas clases e iteradores, sobretodo en las colecciones de clases.

A continuación se muestra la organización de las clases (Figura 6.1):

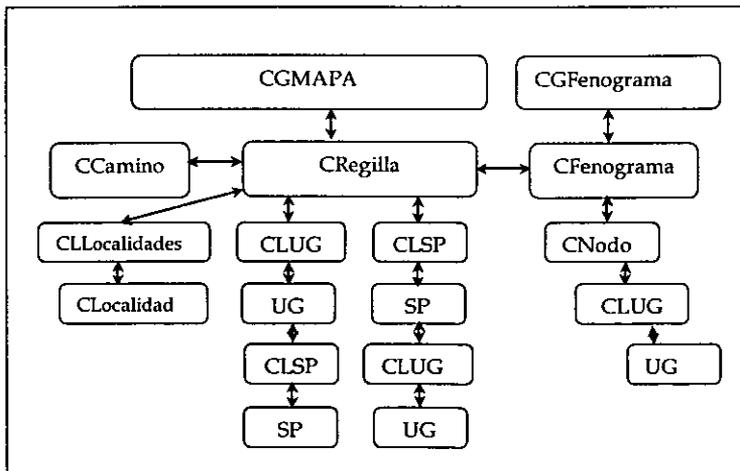


Figura 6.1 Clases básicas detectadas en el primer paso.

El último paso aplicado para esta fase, fue la especificación de las interfaces de las clases, se definieron las funciones públicas y los argumentos así como su tipo, ésta fue la parte más detallada.

No se efectuaron los pasos una sola vez, sino varias, en combinación de la metodología de entrega por etapas. Las iteraciones, correspondieron a una entrega en la metodología :

1. La primera iteración sirvió para el diseño global de los objetos, es decir la arquitectura básica.
2. La segunda iteración, fue más detalla en aquellas clases y sus interfaces necesarias para la implementación de los análisis de la primera entrega (los detectados en la matriz de selección del análisis de requerimientos).
3. La siguiente iteración, fue más detalla en las siguientes clases e interfaces de la siguiente etapa.
4. Y así sucesivamente hasta terminar el software (planeado en 3 entregas).

Se utilizó pseudocódigo en el diseño detallado de los métodos de las clases implementadas en cada iteración posterior al diseño general. Esto permitió que las funciones y métodos de las clases tengan una adecuada modularización del código, y una codificación mucho más sencilla y evitando las sorpresas y riesgos de esta etapa.

Cabe señalar que la arquitectura pensada en un principio cambió, pues algunas clases que se apreciaban como componentes monolíticos, se descompusieron en un conjunto de clases para mejorar modularidad, su tamaño, reducir su complejidad e incrementar el desempeño y eficiencia del código. Un ejemplo de esto es la clase que se definió en el diagrama de objetos como la rejilla, que en realidad es un conjunto de clases cooperando entre sí para operar sobre las colecciones de objetos UGs y especies para realizar todos los análisis de la Biogeografía cuantitativa que se acordaron diseñar e implementar en la etapa del análisis de la aplicación.

En el diagrama anterior no se muestran una serie de clases que apoyan a las clases propias del sistema, estas clases tienen funciones muy particulares y cooperan en general con todas las clases y demás componentes del sistema. En este tipo de objetos se encuentran las siguientes clases:

- Una clase encargada de todo el manejo de la BD del sistema: Esta clase está encargada de abrir, cerrar, leer y escribir en la BD. La comunicación entre objetos es a través de tipos estándares y de *RecordSet's* del entorno de desarrollo (Ver Cáp. 7).
- Clase de control de versión: Es la que permitirá controlar la funcionalidad disponible del sistema ya sea porque aún no ha sido implementada o por alguna razón comercial.
- Clase de manejo de errores: Es la clase encargada del manejo de todos los errores de la aplicación.
- Algunos controles encapsulados en clases: Estas clases permiten agregarle funcionalidad extra a estos controles de manera específica a las necesidades del desarrollo, en este tipo de clases se encuentran los cuadros de simbología y listas de especies.

6.3.1 La Arquitectura Final.

La arquitectura resultante no solo se formó de objetos, sino fue una mezcla de paradigmas, esto se debió a varias razones:

- Limitaciones del lenguaje de desarrollo: En este caso el lenguaje no es totalmente orientado a objetos, y muchas de sus características siguen siendo procedurales.
- Aprovechar las ventajas de los demás Paradigmas: En algunas partes de la aplicación, la orientación a objetos trae consigo código extra, duplicación de código y las consecuencias que ello trae en el desempeño y en el gasto de memoria. Para evitar esto se utilizaron módulos de código tratando de obtener lo mejor de la programación estructurada ya que algunos componentes de la aplicación el diseño de un objeto quedaba muy sobrado para la función que iba a desempeñar (ejm. No se

necesita una clase que se instancie n veces, basta con una sola vez que esté en memoria).

Al igual que la BD, el diseño de la arquitectura del sistema está dividida de manera lógica por lo que dentro del sistema se encuentran:

6.3.1.1 Menús:

Archivo	Descripción
Nueva AE..	Despliega cuadro de diálogo de nueva AE. (navegación entre archivos).
Abrir AE...	Despliega cuadro de diálogo de abrir AE (navegación entre archivos).
Abrir Geoframe...	Cuadro de diálogo de abrir Geoframe. Una vez abierto el AE permite abrir una rejilla y un geoframe asociado, así como un camino metodológico.
Importar Datos	Cuadro de diálogo de importación de datos.
Exportar Datos	Cuadro de diálogo de exportación de datos. (Algunos Análisis, como la matriz de presencia ausencia).
Guardar ...	Guardar camino metodológico. (Cuadro de diálogo cuando sea pertinente).
Guardar como..	Guardar camino metodológico, geoframe y rejilla según el contexto y archivo completo (se debe recalcar que se debe hacer antes esto pues no funciona igual que Word).
Imprimir	Cuadro de diálogo de imprimir.
Salir	Salir del sistema.

Análisis	Descripción
Mapa Alfa	Genera el mapa Alfa en pantalla.
Mapa de Similitud	El ratón cambia de forma para seleccionar la UG pivote del mapa de similitud.
Matriz coloreada	Despliega la matriz coloreada.
Matriz de Similitud	Despliega sub menú de matriz de similitud.
Bioestadísticas	Despliega diálogo de las estadísticas.
Agrupamiento	Despliega diálogo de agrupamiento.
Áreas Prioritarias.	Despliega diálogo del método de Rebelo iteración por iteración.

<i>Edición</i>	Descripción
Nombre de la UG	Despliega diálogo de nombre de la UG (debe haber una seleccionada).
Copiar	Copiar al portapapeles.
Pegar	Pegar desde el portapapeles.

<i>Herramientas</i>	Descripción
Opciones	Despliega diálogo de opciones.
Definir Rejilla	Despliega diálogo de la definición de la rejilla.
Definir Geoframe	Despliega diálogo de la definición Geoframe.
Paletas Y Rangos	Despliega cuadro de diálogo acerca de paleta y rangos a aplicar para cada análisis.
Camino Metodológico	Despliega el cuadro de diálogo de información del camino metodológico.
Selección manual.	Herramienta de seleccionar UG's. (por rango s en el mapa o manual).
Seleccionar por rango	Cuadro de diálogo para selección por algún rango en el mapa.
Seleccionar de toda el AE.	Selecciona toda el AE de la siguiente manera: a) Si solo hay Rejilla selecciona la rejilla. b) Si hay Geoframe selecciona el Geoframe.
Usuarios	Despliega el diálogo de usuarios que permite seleccionar uno o darlo de alta.

<i>Preguntas</i>	Descripción
Localidades	Despliega diálogo de las localidades.
Áreas de Distribución	Despliegan diálogo de las áreas de distribución.
Lista de Especies	
De toda el AE.	Despliega el resultado de especies de toda el AE.
De las UGs seleccionadas	Despliega diálogo de las listas de especies.
UG's por Riqueza	Despliega diálogo UG's por riqueza.
Especies por Distribución	Despliega diálogo especies por distribución.
Similitud entre UG's	Despliega diálogo de la similitud entre UG's.
Limpiar	Regresa al estado inicial (no hay pregunta previa).

Nota: Siempre que se pueda conservar el comando anterior.

<i>Ver</i>	Descripción
Acercar	Herramienta de acercar en el mapa, matriz y fenograma.
Alejar	Herramienta de alejar en el mapa, matriz y fenograma.
Ver retícula	Muestra y oculta la retícula.
Ver reglas	Muestra y oculta las reglas.
desplazamiento	Herramientas de desplazamiento en el mapa, matriz y fenograma.
Toda el AE.	Vista de toda el AE en una pantalla.
Información de la UG	Cuadro de diálogo de información de la UG.

<i>Ventana</i>	Descripción
Casos y Subcasos.	Diferentes vistas de las ventanas de resultados.

6.3.1.2 Formas:

Generales

- Frma_Menus: Es la que contiene la forma principal de la aplicación, y en ella se encuentra la barra de herramientas y la barra de menús.
- Frma_Mapas: Es la que tiene las pantallas de salida que se definieron en la etapa de análisis.

Archivo

- Frma_AbrirGeo Este diálogo permite abrir una rejilla, geoframe y el resultado de un camino metodológico.
- Formas de Guardado: Estas formas son las que permiten salvar las definiciones de la rejilla, el geoframe y el camino metodológico:
 - Frma_GuardarRejilla.
 - Frma_GuardarGeo.
 - Frma_GuardarCamino
- Formas de importación de datos de recolección: Estas formas permiten hacer el mapeo de la fuente de datos a la BD, así como una serie de características que eviten la duplicidad de datos por errores de captura, las formas que componen este bloque son:

- Frma_Importa1
- Frma_Importa2
- Frma_Importa3n
- Frma_Exportar: Este diálogo permite la exportación de datos y resultados del sistema.
- Frma_Imprimir: Este diálogo permite seleccionar lo que se desea tener en papel a través de una impresora.

Edición

- Frma_Nmbre: Permite editar el nombre de la UG.

Ver

- Frma_Info: Muestra información de UG: Especies, diversidad alfa, si esta incluida en el geoframe, puntos geográficos de definición, etc.

Análisis

- Frma_Bios: Muestra los análisis que tienen como salida un número y no dependen de ninguna selección, los cuales son : Alfa promedio, diversidad Beta, diversidad gamma, Estimación de riqueza y calidad del muestreo.
- Frma_Agrpa: Permite configurar los parámetros de los análisis de agrupamiento.
- Frma_Pri: Esta forma interactúa con el usuario para llevar a cabo el análisis de áreas prioritarias.

Preguntas

- Frma_Loc: Permite configurar los parámetros para mostrar un mapa de localidades de recolección.
- Frma_AreaDis: Muestra el diálogo que en función de un conjunto de especies genera un mapa de distribución de especies.
- Frma_ListaSp: despliega un diálogo con la lista de especies de las UG's seleccionadas.

- Frma_UGRqza: Despliega un diálogo que permite configurar el mapa de especies endémicas.
- Frma_DisSP: Despliega un diálogo que permite configurar cuáles especies están en n UG's (especies de distribución restringida):
- Frma_Sim: Despliega un diálogo para realizar el análisis de similitud entre conjuntos de UG's.

Herramientas

- Frma_Opcnes: Diálogo que permite configurar algunos aspectos del sistema, como lo son: el criterio de similitud, de riqueza y la beta a aplicar.
- Frma_Rangos: Diálogo que permite la configuración de los rangos y paletas de colores a utilizar en los mapas resultado.
- Frma_Reglla: Diálogo que define las características geográficas de la rejilla.
- Frma_Geo: Diálogo que define las UG's que serán consideradas en cualquier análisis.
- Frma_Cmno: Diálogo que permite definir un camino metodológico.
- Frma_Usrios: Diálogo de administración de usuarios del sistema.
- Frma_Selccion: Diálogo que nos permite seleccionar UG's por rangos de algún resultado en el mapa.

6.3.1.3 Clases.

1 Clases Gráficas:

- CGMap: Clase que encapsula un control Open GL para el despliegue, manejo e interacción con el mapa gráfico.
- CGFenograma: Clase que encapsula un control Open GL para el despliegue, manejo e interacción con el fenograma gráfico.

2 Clases asociadas con las funciones del sistema:

- CVersion: Clase encargada del control de las operaciones que puede efectuar el Software en función de su versión.

- CImporta: Clase encargada de la importación de nuevos datos a la BD.
- CExporta: Clase encargada de la exportación de datos y resultados a archivos planos.
- CBD: Clase encargada de leer y escribir sobre la BD (a excepción de CImporta y CExporta).
- CError. Clase encargada del manejo de errores.

3 Clases propias a los análisis biogeográficos:

- CUG: Clase que representa una UG.
- CSP: Clase que representa una especie.
- CLocalidad: Clase que representa una localidad de recolección.
- CLLocalidades: Clase lista de localidades.
- CLUG: Clase lista de UGs, esta encargada de agrupar a una o varias UGs.
- CLSP: Clase lista de especies, esta encargada de agrupar una o varias especies.
- CNodo: Clase que representa un nodo del fenograma.
- CFenograma: Clase que se encarga de la construcción y las operaciones propias del fenograma.
- CRejilla: Clase que contiene una serie de listas tanto de UGs como de especies, operando con ellas para poder realizar los análisis requeridos, es en realidad una serie de clases:
 - CRejillaAbre: Clase encargada con todas las operaciones de apertura de la rejilla, geoframe, camino metodológico.
 - CRejillaSimilitud: Clase encargada de la creación y manipulación de la matriz de similitud, así como del mapa de similitud.
 - CRejillaAlfa: Clase encargada de la creación del mapa alfa.
 - CRejillaRiqueza: Clase encargada de todos los cálculos de estimación de la riqueza.
- CCamino: Clase encargada del control del camino metodológico.

6.3.1.4 Módulos.

- Principal: Está encargado de arrancar la aplicación.
- Archivo: Está encargado del manejo del menú archivo.
- Math_Set: Encargado de operación de conjuntos y de operaciones matemáticas varias.

6.4 Plan de Implementación.

Siguiendo con la metodología, el último punto de este diseño general del SI, es la planeación de las etapas, cada una de ellas constará de una fase de diseño más específica de cada parte a implementar, siendo este diseño un trazo general.

En la determinación de qué se implementa primero y qué después, fue de mucha ayuda la documentación que se generó en la fase de análisis en función de la importancia y dependencia de los análisis a implementar, y de las funciones que debe tener el sistema para trabajar con él. A continuación se muestra el plan de implementación:

Archivo	Etapas y Descripción
Nueva AE..	1era etapa: Creación de la BD de tipo access con todo el diseño.
Abrir AE...	1era etapa: abrir BD .
Abrir Geoframe...	1era. etapa: Abrir de la BD la rejilla y el geoframe. 3era. etapa: Abrir el camino metodológico.
Importar Datos	2da etapa: Importación sencilla de datos (Mapeo). 3era. etapa: Revisión de especies , localidades, colectores.
Exportar Datos	3era etapa.
Guardar ...	1era etapa: A excepción de los caminos metodológicos. 2da etapa: Caminos Metodológicos.
Guardar como..	3era etapa.

Imprimir	2da etapa: Matrices, listas y bioestadísticas. 3era etapa: Mapas y fenograma.
Salir	1era etapa.

Análisis	Etapa y Descripción
Mapa Alfa	1era etapa.
Mapa de Similitud	1era etapa: Generación del mapa con los criterios del listado por análisis para esta etapa.
	2da etapa: Análisis del listado para esta etapa.
Matriz coloreada	2da etapa.
Matriz de Similitud	1era etapa.
Bioestadísticas	1era etapa.
Agrupamiento	2da etapa. Generación del fenograma y mapa.
	3da etapa. Mapa dinámico con el nivel de corte del fenograma.
	3era. etapa.
Áreas Prioritarias.	3era. etapa.

Edición	Etapa y Descripción
Nombre de la UG	1er etapa.
Copiar	3era etapa.

Herramientas	Etapa y Descripción
Opciones	1era etapa.
Definir Rejilla	1era etapa.
Definir Geoframe	1era etapa.
Paletas Y Rangos	1era etapa. Operación con ellos.
	3era etapa: Diseño de todos las opciones por omisión en la BD.
Camino Metodológico	3era etapa.
Selección manual.	1era etapa.
Seleccionar por rango	1era etapa.
Seleccionar de toda el AE.	1era etapa.
Usuarios	3era etapa.

Preguntas	Etapa y Descripción
Localidades	1era etapa: Pinta localidades.
	2da etapa. Clasificación.
Áreas de Distribución	2da etapa.
Lista de Especies	1era etapa.
De toda el AE.	1era etapa.
De las UGs seleccionadas	1era etapa.
UG's por Riqueza	2da etapa.
Especies por Distribución	2da etapa.
Similitud entre UG's	3era etapa.
Limpiar	1era etapa.

Ver	Etapa y Descripción
Acercar	1era etapa.
Alejar	1era etapa.
Ver retícula	1era etapa.
Ver reglas	2da etapa.
desplazamiento	1era etapa.
Toda el AE.	1era etapa.
Información de la UG	1era etapa.

Ventanas	Etapa y Descripción
Restaurar	1era etapa.
	2da. Etapa: Control dinámico de los tamaños.
Maximizar mapa.	1era etapa.
Maximizar mapa y simbología.	1era etapa. 2da. Etapa: Control dinámico de los tamaños.
Maximizar lista especies	1era etapa.
Maximizar fenograma.	1era etapa.
Fenograma y Mapa	1era etapa.
	2da. Etapa: Control dinámico de los tamaños.

Ayuda	Etapa y Descripción
Ayuda QBMap	3era etapa.
Acerca de..	3era etapa.

7 Implementación

En esta etapa se detalla la selección de la plataforma de desarrollo, aunque en las etapas anteriores se tenía una visión sobre el tipo de herramientas con las que se podía contar; algunos problemas que se tuvieron que resolver antes de empezar la codificación del SI propiamente dicho en su primera etapa y los pasos seguidos para implementar esta primera etapa.

En la etapa de diseño ya se sabía en general qué plataforma de desarrollo se iba a usar, algunos aspectos del diseño quedaron pendientes por determinar las herramientas a utilizar, por lo que el diseño no fue dirigido hacia ninguna herramienta en particular, y en el comienzo de la implementación se evaluaron diferentes alternativas, y se pusieron metas de solución de problemas de desarrollo, para saber si se utilizaba o no.

7.1 Plataforma de Desarrollo.

La plataforma de desarrollo se refiere a las herramientas que se utilizaron para la implementación del SI, que van desde el lenguaje empleado, hasta componentes utilizados, que por su importancia son resaltados en esta parte.

7.1.1 El Lenguaje de Desarrollo:

El lenguaje de desarrollo en este caso fue Visual Basic 5.0 de Microsoft (VB), debido a las siguientes ventajas que ofrece el lenguaje:

1. La experiencia en este lenguaje por parte del desarrollador es bastante amplia, en grado de que es el lenguaje que más domina (desde sus versiones interpretadas como lo era GW Basic).
2. La versión 5 del compilador ya genera código nativo para máquinas de 32 bits de la familia de Intel (80X86, Pentium X y equivalentes), ya que

las versiones anteriores de VB generaban un código intermedio denominado código-p, lo que significaba la necesidad de realizar una llamada al motor en tiempo de ejecución, afectando en el desempeño. Una pequeña comparación entre compiladores se puede encontrar con Guy Eddon y Henry Eddon en su libro "ActiveX Visual Basic 5.0" donde se muestra un experimento con una misma función en C y en VB. La función utilizada es la de determinación de si un número es primo, y se usó para encontrar los números primos en el rango de 1 a 50 000. Se muestra la tabla de resultados que obtuvieron :

Tipo de Código	t de ejecución [seg.]	Veces más rápido con respecto al código-p
Dentro del Entorno de Desarrollo de VB	46.25	58%
Código-p	28.86	100%
Código VB nativo Compilado	13.09	205%
Código C compilado	12.8	221%

Como se puede ver el código nativo que genera VB es muy cercano al generado por C, y en nuestro caso el desempeño no es una variable vital (como lo sería en un sistema de tiempo real).

3. La versatilidad que tiene el lenguaje para incorporar componentes extras, que en este caso fue muy útil, pues la parte de graficación del mapa, se utilizó un control de terceros, del cual se hablará más adelante, el cual simplificó, aceleró el desarrollo y mejoró el desempeño del sistema.
4. De los 3 requisitos básicos para que un lenguaje pueda decirse que es orientado a objetos (encapsulación, herencia y polimorfismo) VB soporta 2: La encapsulación y el polimorfismo, la herencia no es soportada por el lenguaje. Pero en nuestro diseño no utilizamos ninguna relación de herencia "es un", por lo que podemos seguir con nuestro diseño de objetos implementado en VB.

5. Independientemente del conocimiento del programador en el lenguaje, VB ha demostrado mayor rapidez en el desarrollo que otras plataformas debido a su simplicidad.

Una de las ventajas que ofrece la orientación a objetos es la reutilización de componentes, y utilizar partes ya hechas para la construcción de algún SI, evitando así el costo de desarrollar una parte determinada del sistema. En este caso se pensó en utilizar controles (componentes) para el manejo geográfico del sistema, entre los que se consideraron fueron:

- Map X de MapInfo Co.: Es un control que trae implementado ya todas las operaciones gráficas, enlace a datos, proyecciones de la tierra, creación de objetos gráficos (puntos, líneas, polígonos y rasters), mapas temáticos, abre nativamente los archivos de MapInfo, etc.
- DBCad de ABACO srl.: Es un control y un conjunto de librerías que al igual de MapInfo trae consigo un conjunto de funciones tales como: las operaciones gráficas, enlace a datos, proyecciones de la tierra, creación de objetos gráficos (puntos, líneas, polígonos y rasters), importación de varios formatos gráficos como DWG y DXF de Autocad V14, ESRI Shapefile, WMF, TIF, BMP, RLE, RLC, etc.
- MapObjects de ESRI.: Es una colección de componentes para desarrolladores de sistemas GIS, la versión 2 trae más de 45 componentes para el mapeo, entre las cuales se pueden mencionar: cambios de proyección, soporte a diversos tipos de datos, geocodificación de información, etc.

~~----- Proyecto Galileo de Hipparchus programmer's workbench program.~~

Galileo es un entorno de desarrollo el cual permite la manipulación directa de datos usando las funciones de las librerías de Hipparchus. Galileo puede ser programado por medio de scripts o haciendo

llamadas de entrada /salida en tiempo de ejecución desde ANSI C/C++.

Cada una de las opciones se analizaron con cuidado, revisando las ventajas y desventajas que tenían, en algunos casos se probaron versiones de prueba ofrecidas por el fabricante, pero la razón de no utilizar ninguna de ellas fue que los recursos que se tenían para el desarrollo eran muy limitados, y aunque el uso de estas herramientas comerciales aceleraban el proceso y podrían agregar funcionalidad al desarrollo, requerían una inversión con la cual no se contaba.

Por lo anterior se decidió crear el motor gráfico de la aplicación, pero ahora sí se utilizó un componente ya hecho: Un control OCX para VB de Open GL (Open Graphical Language).

7.1.2 Open GL.

Open GL es estrictamente definido como una interfaz en software hacia el hardware de gráficos. En esencia, es una librería para gráficos en 3D y modelación, la cual es muy rápida y portable. Open GL utiliza algoritmos que fueron cuidadosamente desarrollados por Silicon Graphics Inc. (SGI), empresa conocida mundialmente como líder en gráficos por computadora y animación.

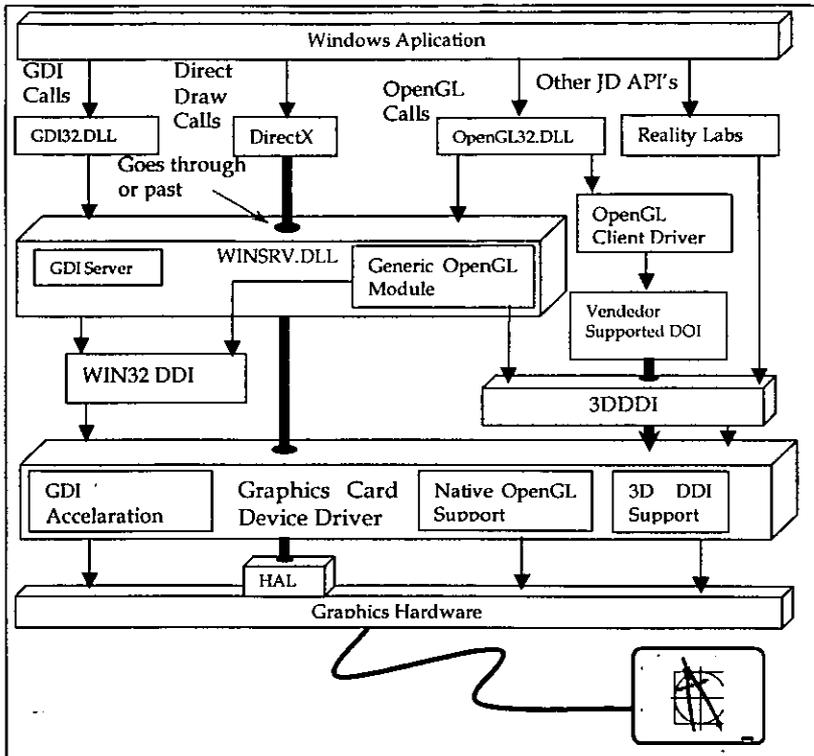
Open GL es propuesto para ser utilizado por computadoras cuyo hardware haya sido diseñado y optimizado para manejo y visualización de imágenes en 3D. Aunque existen implementaciones solo en software de manera genérica para diversas plataformas, tal es el caso de algunas utilizadas en Windows 9X y NT.

7.1.2.1 *Un poco de historia de Open GL y su competencia.*

Open GL es relativamente un nuevo estándar en la industria. El antecesor más remoto provino de SGI como "IRIS GL" el cual era un API (Application Program Interface) de programación para las estaciones de trabajo de SGI, las cuales no eran de propósito general, sino poderosos equipos para el manejo de imágenes.

Cuando SGI enfoca su trabajo a obtener un IRIS GL exportable a otras plataformas, es cuando Open GL nace ofreciendo el poder del GL de SGI pero de manera abierta para su fácil adaptación a diverso hardware y sistemas operativos.

En julio de 1992 la versión 1.0 de las especificaciones de Open GL son liberadas, y Microsoft promete soporte en su sistema operativo Windows NT 3.51, compromiso que rompería posteriormente.



El principal problema que tenía el ambiente Windows de Microsoft para el manejo de gráficos es que no se podía manejar directamente al hardware de video, y se pasaba por una cascada de llamadas que afectaban gravemente el desempeño, pero los riesgos de manejar directamente el hardware son muy altos:

- Cada chip tenía sus propias librerías propietarias e incompatibles entre sí, es decir para cada chip se debía tener un código especial para manejarlo.
- El sistema operativo puede volverse muy inestable, si la aplicación que está controlando hardware tiene problemas.

La manera de resolver esto es crear una única librería que maneje al hardware y que esté bien integrada al sistema operativo, ofreciendo además de que si una función no está soportada por el hardware, se simulará por software, pero si estuviera presente se utilizaría la del hardware, teniendo como resultado este manejo un mejoramiento en el desempeño. Obviamente las funciones definidas deben ser un estándar.

Microsoft quería crear este estándar e integrarlo en Windows 95 y su nueva versión de NT (4.0), y lo nombró DirectX, es en este punto donde rompe su compromiso con Open GL, pese a que lo había utilizado como reclamo tecnológico en Windows NT 3.51, y adquiere una API de Reality Lab 3D, renombrándola como Direc3D.

Otro competidor fue Autodesk, quien creó Heidi para su programa de 3D Studio Max, pero pronto adopta tanto Direct3D como OpenGL.

De aquí en adelante se sucede una guerra por demostrar cuál era el mejor: Microsoft libera sus versiones 3.0, 5.0 y 6.0 de sus Direct3D, y Open GL pasa a la versión 1.2 por un desarrollo propio de SGI con el apoyo Intel, Digital, HP, IBM y Evans & Sutherland.

7.1.2.2 El papel de Open GL en el proyecto.

Como ya se mencionó, Open GL se utiliza a través de un control OCX de VB para el manejo del mapa dentro de nuestra aplicación. Las metas propuestas para la utilización de este componente fueron:

- Crear rectángulos en función de 2 puntos.

- Las operaciones gráficas de acercamiento, alejamiento (Zoom's) y cambios de vistas (pan) en 2D.
- Selección de elementos gráficos.

Se probaron 2 controles:

- WaiteGL.ocx: Proporcionado por Richard S. Wrigth y Wichel Sweet en su libro "Open GL Superbible, The complete Guide to Open GL Programming for Windows NT and Windows95", el cual no se utilizó pues falló la meta de selección de elementos gráficos.
- GlxCtl.ocx: Encontrado en Internet en la dirección www.hk.super.net/~edx el cual es una librería que soporta las rutinas de Open GL. Es un control ocx que maneja los parámetros iniciales para Open GL y dispara varios eventos para controlar la aplicación que son procesados antes de GLUT (librerías fuera de Open GL, pero diseñadas específicamente para cada plataforma para el manejo de las funciones, los eventos del mouse y teclado). Cumpliéndose con este componente las metas propuestas con sobrada funcionalidad.

Las ventajas dentro del proyecto del uso de este control son: el desempeño del manejo gráfico es muy aceptable; no se tuvo que programar el manejo del mapa (qué se ve, cómo se ve y qué no se ve en una vista determinada), pues el control se hace cargo de esto; el manejo de los colores para los temáticos y una intuitiva interfaz de point a click que es unos de los objetivos de la aplicación.

7.1.3 La Base de Datos:

La BD utilizada por este desarrollo fue el motor de la librería DAO 3.6 de Microsoft, el mismo utilizado para su producto Access por las siguientes razones:

- El costo para el usuario final, pues uno de los manejadores más fáciles y baratos de conseguir en el mercado.

- Fácil utilización de ODBC (Open Data Base Connect), lo cual será implementado en la 2da versión, siendo mínimo el costo para hacerlo.
- No se considera que la BD vaya tener una carga muy fuerte de transacciones de escritura por una cantidad grande de usuarios simultáneamente, por lo que lo ofrecido por este motor en este aspecto es suficiente.
- Soporta la mayoría del SQL estándar, y la totalidad de lo que necesitamos de SQL.
- El tamaño límite de una tabla de este motor es de alrededor de 1 GB, no se cree que alguna tabla del sistema llegue a tal tamaño.

Sobre la creación de la BD se tienen 2 opciones por considerar, de las cuales aún no se decide sobre una en particular, pues en esta primera etapa de desarrollo no es de vital importancia :

- El copiado de una plantilla: Dicha plantilla contendrá tanto el diseño como los datos iniciales que necesita la aplicación (rangos y colores). La ventaja es que esto es muy fácil y rápido de implementar pero con la desventaja de que se trabajará doble pues para crear la estructura en una BD diferente a través de ODBC se tendrá que implementar la segunda manera.
- Creación de la estructura mediante un script SQL: El cual estará dentro de la aplicación o en un archivo aparte. La ventaja es la flexibilidad para crear la estructura de la BD en cualquier manejador, su desventaja es que los datos con los que debe de arrancar la aplicación deben ser leídos de la plantilla.

7.2 El Desarrollo.

El primer paso fue cumplir las metas de graficación que se mencionó en la parte de Open GL, las cuales fueron encapsuladas en una clase tal y como lo indicaba el diseño, resaltando que la arquitectura permitirá en manejo de capas de diferente información: ya sea geográfica (toponimia, divisiones municipales, ríos, lagos, etc) o cosmética (rosa de los vientos, reglas, marcos, etc). Esta clase sólo se encarga del manejo de Open GL, cada ente gráfico es responsable de cómo se dibuja así mismo. Es en esta parte donde se manifiestan las ventajas de usar objetos y en específico del polimorfismo como se muestra en este código:

```
Public Sub Dibuja()
'FRS 21/Sep/1999
'Procedimiento: Dibuja el mapa, dependiendo del orden de dibuja uno arriba
de otro
Dim ObjetoGrafico As Object
.....
'Ciclo que pinta capas y elementos
For i_cont = 1 To i_NCapa
'si es la capa de UG's pinta cada elemento de la colección de UG's
If i_cont = i_CapaUG Then
For Each ObjetoGrafico In C_UGs
ObjetoGrafico.Dibuja
Next
Else
'si es cualquier otra capa pinta cada elemento de la colección capa
For Each ObjetoGrafico In C_capa(i_cont)
ObjetoGrafico.Dibuja
Next
End If
Next i_cont
...
End Sub
```

Y posterior a esto, se levantó toda la interfaz de la aplicación:

- De donde se toman todos los mensajes: menús, cuadros de diálogos, etc.
- ¿Qué controles y cuales son los iconos a utilizar en la barra de tareas?, ¿en qué momento se cargan?, ¿cuándo pueden ser utilizados y cuándo no?.

- Los cursores a utilizar según la herramienta gráfica utilizada (zoom, pan y selección).

A continuación se programaron la apertura de la BD, la apertura de la rejilla, del geoframe, la creación del mapa, la lista de especies de todo el geoframe, la simbología, la información de una UG, la edición del nombre de la UG y las opciones del sistema. Para cada uno de estos aspectos se terminó toda su lógica de operación, en el caso de rangos y paletas solo se implementó en esta etapa la selección de ellos para los análisis implementados.

Dentro de las operaciones propias del sistema se implementaron: el mapa alfa con los rangos de intervalos constantes y manual, todos los índices de similitud y la creación del mapa de similitud con los rangos de intervalos constantes y manual, las bioestadísticas: alfa promedio, beta, gama, casi todos los estimadores de riqueza y la calidad de muestreo.

7.3 La Documentación.

Este aspecto del desarrollo se cuidó mucho el comentar el código y seguir con los lineamientos propuestos en la etapa del diseño sobre la codificación de la aplicación: nomenclatura de variables, manejo de errores, tipo de comentarios, etc. Este tipo de comentarios dentro del código son de gran ayuda en el mantenimiento y evolución del sistema, máxime si son terceras personas quienes realizan esto.

No solamente la documentación se refiere a comentar el código, sino a un conjunto de escritos que son más detallados que los del diseño y son resúmenes de qué y cómo fueron implementadas las funciones del sistema. Por cada componente del sistema se tienen un documento específico al tipo de elemento que es:

- Formas: Para todas las pantallas del sistema implementadas en esta etapa se cuenta con un documento en el que se detalla su funcionalidad, la imagen final de ella y la fuente de todos sus mensajes al usuario (etiquetas, mensajes de foco, mensajes de error, etc.)

- Módulos: Cada módulo cuenta con un documento que detalla todas sus funciones y procedimientos públicos.
- Clases: Estos componentes tienen una documentación un poco más amplia, de ellos se detalla su función, sus propiedades, sus funciones y métodos públicos. Esta documentación es similar a la del diagrama de clases, sólo que un poco más detalla.

La documentación aquí mencionada se refiere al diseño detallado que se debe hacer en esta etapa según la metodología, anterior al desarrollo. Y los productos generados son el sistema y una carpeta que aglutina todos los documentos del sistema, desde su análisis, diseño y desarrollo hasta esta etapa.

7.4 Las Sigüientes Etapas

Tal y como lo marca la metodología, al finalizar esta primera etapa se tendrá un sistema con la funcionalidad propuesta, que puede ser utilizado por los usuarios, recalando que todavía no es un sistema totalmente terminado, que se puede trabajar sobre él pero no tendrá toda la funcionalidad esperada del sistema final, algunas operaciones se deberán hacer por fuera y otras tantas serán inestables.

El siguiente paso será hacer el diseño detallado de los aspectos a implementar: nuevas clases; agregación de propiedades funciones y métodos a las ya existentes; documentación de las formas y módulos a utilizar. Y una vez terminado esto implementar la 2da etapa. De manera similar la 3era y última etapa, teniendo como resultado:

- El sistema final: con toda la funcionalidad prevista y un programa de instalación que permita su fácil distribución.
- La carpeta de proyecto: que contenga todos los documentos generados desde el análisis, diseño y desarrollo por etapas.

- y los manuales para el usuario final:
 - Manual del usuario: éste contendrá el concepto del sistema, es decir cual es su uso (análisis biogeográficos), sus alcances, su instalación y requerimientos de equipo, la manera de trabajar con él, empezando por lo más simple e indispensable hasta lo más complejo del sistema, un ejemplo de trabajo y por último un listado de la funcionalidad del sistema.
 - Manual Técnico: éste contendrá la teoría biogeográfica que lo respalda (formulas y conceptos), el modelo entidad relación de la BD, el diccionario de datos, y aspectos muy básicos de la arquitectura del sistema (como puede ser la relación de UG y especies).

Una vez liberado el sistema la fase que le sigue es la de control de errores, y dependiendo de su desenvolvimiento observado entre los usuarios, se tomarán los aspectos resaltados en este desarrollo y los propuestos por los usuarios para mejorar el sistema en una segunda versión.

8 Conclusiones.

El objetivo de la presente tesis se cumple con el análisis y diseño completo del Sistema de Información Geográfica propuesto para el área de la Biogeografía Cuantitativa, se presenta de manera agregada la implementación en un 70% de la primera etapa como ejemplo de lo que será el sistema final.

Con este desarrollo de la presente tesis, se provee a los investigadores de esta rama de la ciencia con una herramienta que les hará el trabajo más fácil. Pero se reconoce que esta herramienta SIG, como cualquier otra, es meramente servidora de la geografía cuantitativa, y no al revés, es decir, el SIG -como el teléfono y el correo electrónico- es una herramienta que ayuda a hacer el trabajo un poco más fácil, más completo y en el mejor de los casos más provechoso.

Esta herramienta permitirá al biogeógrafo cuantitativo estudiar a nuestro planeta bajo su campo de estudio, ayudándolo en su tarea diaria de recolectar, organizar y analizar información de la flora y fauna de una región en específico para poder tomar decisiones para proteger áreas, o simplemente para caracterizar un área o una especie en específico. Es responsabilidad de estos investigadores no perderse en el manejo de estas herramientas, sino aprovecharlas para que les proporcione información oportuna, de mayor calidad con resultados rápidos y precisos para la toma de decisiones o conclusiones que se deban de hacer.

Desde el punto de vista de la Ingeniería en Computación, la presente tesis permitió aplicar una amplia gama de materias aprendidas durante el transcurso de la carrera, ampliando y reforzando estos conocimientos, así como la investigación: desde la más práctica como puede ser el tipo de herramientas a utilizar, así como de conceptos, al adentrarse en áreas nuevas: Geografía y Biología.

Espero, sinceramente, que esta contribución nos ayude a convivir con la naturaleza de una mejor manera, aprovechando razonadamente lo mejor de ella, pero sin destruirla como lo hemos venido haciendo, dejando a nuestra descendencia una vasta riqueza natural, cultural y tecnológica.

9 Apéndice 1: Notaciones de Generales.

9.1 Notación de las Bases de Datos:

1. Los nombres de tablas empezarán con el prefijo QB_ a su nombre (ej. QB_SP).
 - 1.1. Los nombres de las tablas serán abreviaciones en inglés.
 - 1.2. Sus nombres serán en singular.
 - 1.3. Los nombres de las tablas serán en mayúsculas.
2. Las tablas que sirvan de relación tendrán el prefijo QR_ a su nombre, el cual será herencia de las tablas que relaciona (ej. QR_SP_LOC).
3. Los campos llaves tendrán un prefijo id_ y el nombre será el mismo que el de la tabla donde son llave primaria.
4. Los campos de cadena tendrán el prefijo s_.
5. Los campos numéricos llevarán el prefijo n_ para los enteros y f_ para los flotantes.
6. Los campos booleanos tendrán al prefijo b_.
7. Los nombres de los campos serán en minúsculas.

9.2 Notación de los Controles y Formas:

1. Todos los nombres serán con la 1ª letra mayúscula en su prefijo, para facilitar la depuración, ya que VB no es sensible, pero cambia automáticamente a como fue definido el control o forma.
2. A los nombres se les eliminarán todas las vocales a excepción que empiecen con una de ellas y la última.
 - 2.1. Se pueden hacer algunas excepciones con nombres muy largos y utilizar abreviaciones.
3. Las formas llevarán el prefijo Frma_ , y el nombre lógico es idéntico al nombre del archivo que la contiene.
4. Los controles del tipo *combo* llevarán el prefijo Cmbo_.
5. Los controles del tipo *botón* llevarán el prefijo Btn_.
6. Los controles del tipo *Label* llevarán el prefijo Lab_.
7. Los controles del tipo *Cajas de texto* llevarán el prefijo Txt_.
8. Los controles del tipo *radio* llevarán el prefijo Rdo_.
9. Los controles del tipo *check* llevarán el prefijo Chk_.
10. Los controles del tipo *marco* llevarán el prefijo Mrco_.
11. Los controles del tipo *flexgrid* llevarán el prefijo Flexs_.
12. Los controles del tipo *increment-decrement* llevarán el prefijo Incr_.
13. Los controles del tipo *barra de avance* llevarán el prefijo Avnce_.
14. Los controles del tipo *picture* llevarán el prefijo Pctre_.
15. Los controles del tipo *menus* llevarán el prefijo Mnu_.
 - 15.1. En caso de los menús heredarán el nombre del padre (sin el prefijo).
16. Los controles del tipo *barra de mensajes* llevarán el prefijo Msg_.
17. Los controles del tipo *barra de desplazamientos* llevarán el prefijo ScrBar_.
18. Los controles del tipo *Lista de Imágenes* llevarán el prefijo Img_.
19. Los controles tipo *CommonDialog* llevarán el prefijo Com_.
20. Los controles tipo *Frame* llevarán el prefijo Fme_.
21. Los controles tipo *GLctl* llevarán el prefijo GL_.
22. Los controles tipo *UpDown* llevarán el prefijo Upd_.

9 Apéndice 1: Notaciones de Generales.

9.1 Notación de las Bases de Datos:

1. Los nombres de tablas empezarán con el prefijo QB_ a su nombre (ej. QB_SP).
 - 1.1. Los nombres de las tablas serán abreviaciones en inglés.
 - 1.2. Sus nombres serán en singular.
 - 1.3. Los nombres de las tablas serán en mayúsculas.
2. Las tablas que sirvan de relación tendrán el prefijo QR_ a su nombre, el cual será herencia de las tablas que relaciona (ej. QR_SP_LOC).
3. Los campos llaves tendrán un prefijo id_ y el nombre será el mismo que el de la tabla donde son llave primaria.
4. Los campos de cadena tendrán el prefijo s_.
5. Los campos numéricos llevarán el prefijo n_ para los enteros y f_ para los flotantes.
6. Los campos booleanos tendrán al prefijo b_.
7. Los nombres de los campos serán en minúsculas.

9.2 Notación de los Controles y Formas:

1. Todos los nombres serán con la 1ª letra mayúscula en su prefijo, para facilitar la depuración, ya que VB no es sensible, pero cambia automáticamente a como fue definido el control o forma.
2. A los nombres se les eliminarán todas las vocales a excepción que empiecen con una de ellas y la última.
 - 2.1. Se pueden hacer algunas excepciones con nombres muy largos y utilizar abreviaciones.
3. Las formas llevarán el prefijo Frma_ , y el nombre lógico es idéntico al nombre del archivo que la contiene.
4. Los controles del tipo *combo* llevarán el prefijo Cmbo_.
5. Los controles del tipo *botón* llevarán el prefijo Btn_.
6. Los controles del tipo *Label* llevarán el prefijo Lab_.
7. Los controles del tipo *Cajas de texto* llevarán el prefijo Txt_.
8. Los controles del tipo *radio* llevarán el prefijo Rdo_.
9. Los controles del tipo *check* llevarán el prefijo Chk_.
10. Los controles del tipo *marco* llevarán el prefijo Mrco_.
11. Los controles del tipo *flexgrid* llevarán el prefijo Flexs_.
12. Los controles del tipo *increment-decrement* llevarán el prefijo Incr_.
13. Los controles del tipo *barra de avance* llevarán el prefijo Avnce_.
14. Los controles del tipo *picture* llevarán el prefijo Pctre_.
15. Los controles del tipo *menus* llevarán el prefijo Mnu_.
 - 15.1. En caso de los menús heredarán el nombre del padre (sin el prefijo).
16. Los controles del tipo *barra de mensajes* llevarán el prefijo Msg_.
17. Los controles del tipo *barra de desplazamientos* llevarán el prefijo ScrBar_.
18. Los controles del tipo *Lista de Imágenes* llevarán el prefijo Img_.
19. Los controles tipo *CommonDialog* llevarán el prefijo Com_.
20. Los controles tipo *Frame* llevarán el prefijo Fme_.
21. Los controles tipo *GLctl* llevarán el prefijo GL_.
22. Los controles tipo *UpDown* llevarán el prefijo Upd_.

9.3 Notación de Programación y Lineamientos de Documentación.

1. Cada función, módulo, objeto y subrutina deberá tener al inicio, las iniciales del programador y la fecha en que inició la programación de ese bloque al inicio del código.
2. Al final del bloque de código deberá contener la fecha de terminación.

Ej.

Function Cuadrado(numero as integer) as integer

'FRS 01/may/1999

DIM ...

....

'01/jun/1999

End Function.

3. Deberá haber un breve comentario de la función que desempeña ese bloque de código.
 - 3.1. En caso de recibir parámetros se especificará su función y la variable será precedida por la letra p_.
 - 3.2. En caso de devolver valores se especificara que es lo que regresa.
4. Cada bloque de código deberá de definir las variables a utilizar y su tipo al principio del código (Option Explicit).
5. La notación de los prefijos para las variables locales, es como sigue (sólo para variables definidas en el módulo, los parámetros se quedan con su prefijo p_):
 - 5.1. i_ para enteros.
 - 5.2. l_ para enteros largos.
 - 5.3. d_ para dobles.
 - 5.4. f_ para singles
 - 5.5. s_ para cadenas.
 - 5.6. C para clases.
 - 5.7. C_ para el objeto instanciado.
 - 5.8. b_ para booleano.
 - 5.9. v_ para Variant.
 - 5.10. La primera letra después del subguión deberá ir en mayúsculas, para efectos de depuración.
6. El código deberá de estar indentado.
7. En los IF THEN, FOR NEXT, WHILE y SWITCH deberá haber un comentario que explique la condición y el resultado en caso de ser verdadera.
 - 7.1. De ser posible se utilizarán condiciones positivas (evitar el uso de not).
8. Después de las definiciones de las variables deben existir las secuencias de código encargadas de capturar los errores para hacer llamada del objeto de atención de errores, aunque en la etapa de depuración puede estar comentado para facilitar esta actividad.
 - 8.1. En caso de ser errores provocados o cuyo manejo deba ser local, se permitirá código que resuelva el error, en cualquier otro caso se deberá llamar al objeto encargado de su manipulación.
9. En caso de variables globales se utilizará el prefijo g<tipo_variable_ya_comentado>_ deberá existir un comentario de su razón de ser (ejm: gs_nombre variable global tipo string).
 - 9.1. Sólo se podrán definir variables globales en el módulo de principal
10. En caso de variables miembro de un objeto, deberá de haber un comentario de su razón de ser en el punto de definición.
11. Se evitará en lo posible la utilización de las propiedades por default que ofrece el compilador para permitir un código más legible.
12. No se permite el uso de dos o más instrucciones por línea de código.
13. En caso de ruptura anticipada del código (exit), se deberá proporcionar un comentario de su porqué y en que condición.

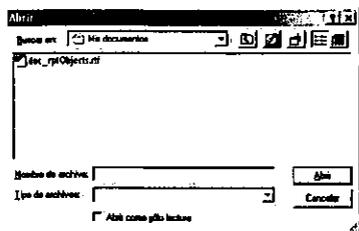
14. Se tratará, en medida de lo posible, el tener la mejor modularización del código.
15. En el encabezado de las clases deberá existir un comentario indicando la lista de objetos de los que depende.
16. Siempre sensible a la tecla ESC, para cancelar el proceso actual.
17. Definir desde el inicio el estado del cursor.
18. Validar toda captura, deshabilitar botones si no se cumple la validación.

10 Apéndice 2: Diagrama de navegación

10.1 Archivo:

Nota: Guardar es un menú que trae un cuadro de diálogo que depende del contexto, se aclara en el menú herramienta.

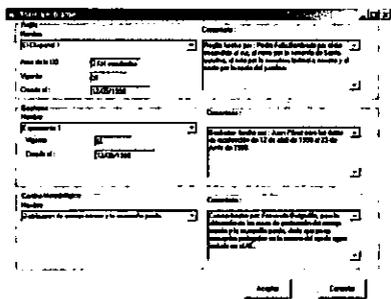
- Nuevo AE y Abrir AE:



Trae el cuadro de diálogo del tipo *Common Dialog* (como el que se muestra), que permite navegar entre los archivos para crear o abrir una Base de Datos de QB-MAP.

- No hay entradas previas.
- No hay análisis.
- Da como salida un mensaje de AE abierta o error en caso de abrir .
- Da como resultado una BD datos completa de QB-Map en el caso de Nuevo.

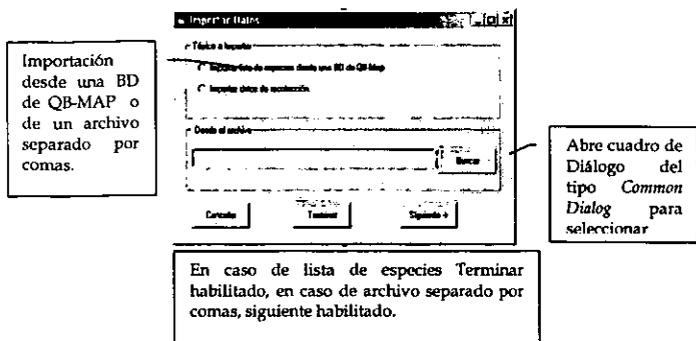
- Abrir Geoframe:



Cuadro de diálogo que se muestra, que permite abrir una rejilla, geoframe y el resultado de un camino metodológico.

- No hay entradas previas.
- Da como salida una salida tipo I.
- Puede haber mapa alfa o resultado del tipo I y II.

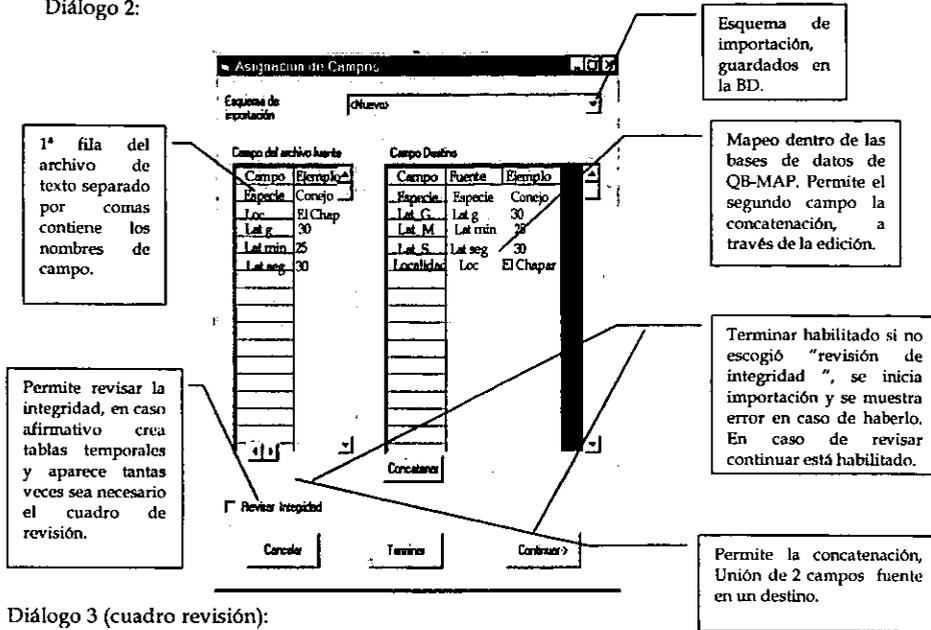
- Importar Datos.



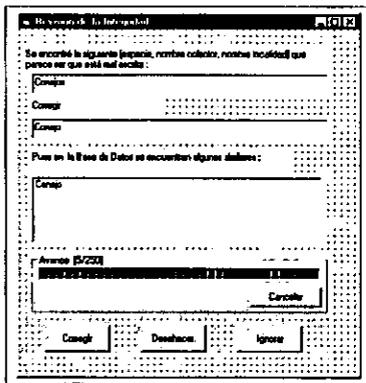
- Debe de estar abierta una rejilla, geoframe o camino metodológico.
- Da como salida un mensaje de importación exitosa o error en la importación.
- Es en varios pasos dependiendo de las opciones que se escogen.

Diálogo 1

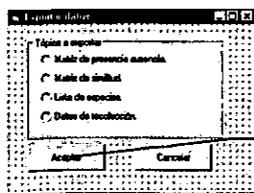
Diálogo 2:



Diálogo 3 (cuadro revisión):



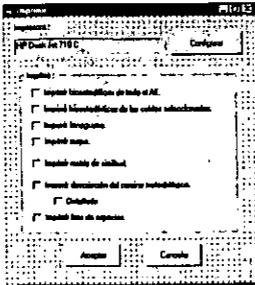
• Exportar Datos.



Al presionar aceptar se abre un cuadro de diálogo de del tipo *Common Dialog* para seleccionar el archivo destino.

- Debe de haber una rejilla, geoframe o camino metodológico abierto.
- Da como salida un archivo separado por comas, con separador de regiones.

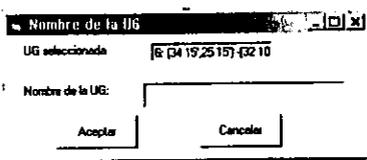
- Imprimir.



- Es sensible al contexto, pues las opciones se habilitan y deshabilitan según lo que haya en la forma principal.
- Da como salida una impresión en papel.

10.2 Edición.

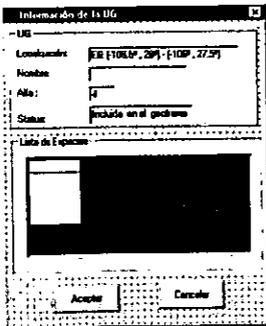
- Nombre de la UG.



- Debe de haber seleccionado, una sola UG.

10.3 Ver.

- Información de la UG.

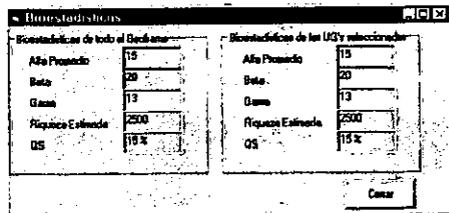


- Debe de haber una UG seleccionada, ya sea por selección normal con "seleccionar", o aparece directamente al presionar el botón de la barra de herramientas de información y dar un click en la UG deseada, de esta manera se puede pedir información de una UG que no está incluida en el Geoframe.
- Se puede pedir información de UG's fuera AE.
- Utiliza el análisis de alfa y lista de especies por UG.

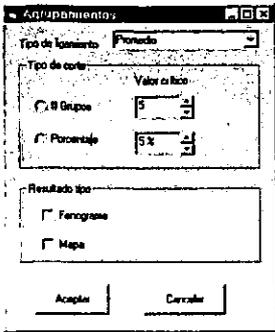
10.4 Análisis.

- bioestadísticas

- Debe estar cargado un geoframe.
- Si hay más de una celda seleccionada se muestran las bioestadísticas del área seleccionada.
- Utiliza los análisis, que se muestran.
- Depende del diálogo de opciones para seleccionar los índices a utilizar (para Beta y riqueza estimada).

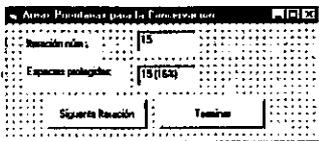


• Agrupamientos.



- Debe haber un geoframe abierto.
- Genera el mapa con el fenograma y sus subcasos.
- El nivel crítico se puede modificar con un control de desplazamiento del fenograma.

• Áreas Prioritarias.



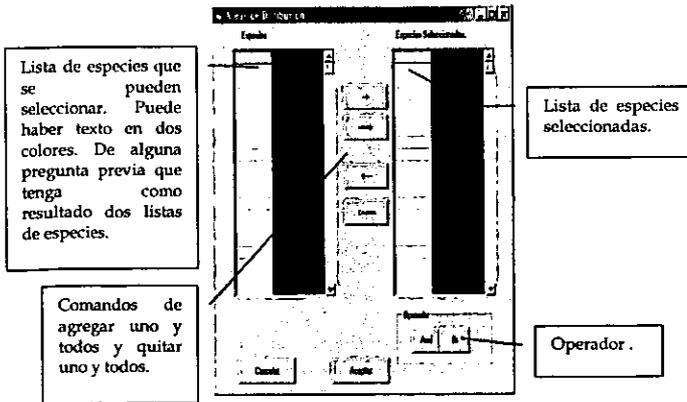
- Debe de haber un geoframe abierto.
- Genera un mapa con UG protegidas por iteración.
- Es iterativo, se corre paso a paso. Preguntas.

• Localidades.



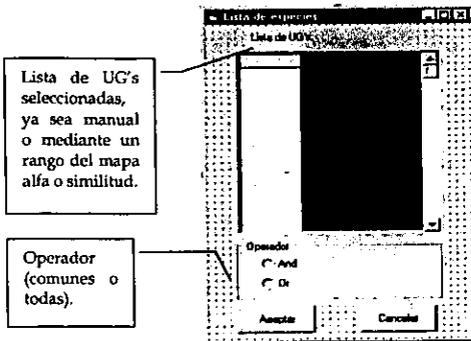
- Se permite este análisis aunque no se haya definido el geoframe, pero sí la rejilla.
- Se le puede empalmar el mapa alfa.
- Muestra un mapa de localidades.

• Áreas de Distribución.



- Debe de haber un geoframe abierto.
- Las especies son de todo el AE o de un análisis previo que haya dado como resultado especies.
- Muestra un mapa de UG's.

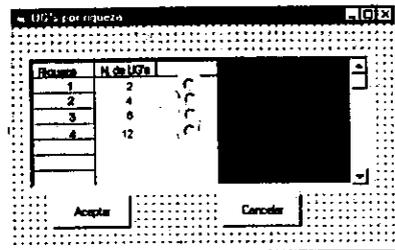
• Lista de Especies.



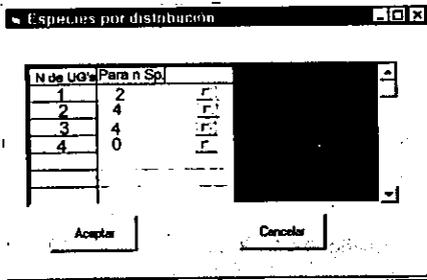
- Se puede pedir una lista de especies de:
 - Un mapa por rangos (análisis previo).
 - Una selección manual previa de UG's.
- Genera una lista de especies con un mapa resaltando las UG seleccionadas para el análisis.
- Con el menú de "ver" se puede "seleccionar toda el AE", y al pedir este análisis se tendrá la lista de especies de toda el AE.

• UG's por Riqueza.

- Debe de haber un geoframe cargado, o algunas UG seleccionadas.
- Da como salida un mapa de UG's de 3 colores.
- Solo se puede seleccionar un renglón.



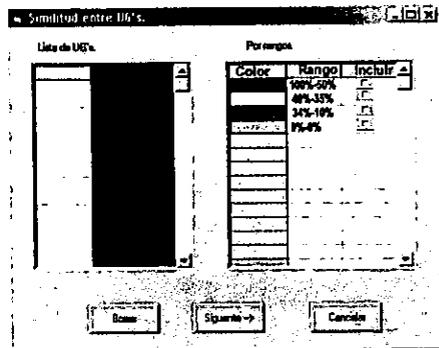
• Especies por Distribución.



- Debe de esta abierto un Geoframe.
- Genera un mapa y una lista de especies de cuáles especies están en n UG's.

• Especies de Distribución Similitud entre UG's.

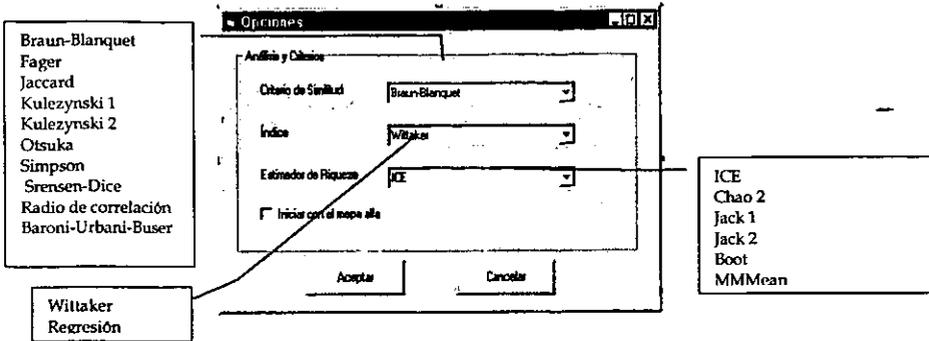
- Debe de haber una selección de dos conjuntos de UG's, por lo que debe haber una selección previa y este cuadro permitirá la segunda selección.
- Se debe poder minimizar y hacer una selección manual con el ratón.
- Se puede seleccionar por rangos dentro de la misma forma, por lo que debe de haber un mapa ya generado.
- Resultado: Numero real y una lista de especies comunes y totales (Unión e Intersección).



10.5 **Herramientas.**

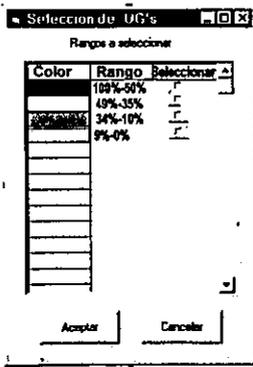
- **Opciones.**

➤ Permite seleccionar los índices a utilizar en los análisis.



➤ Debe de haber un geoframe cargado.

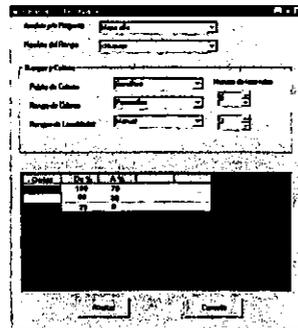
- **Seleccionar UG por rangos.**



- Debe de haber un mapa de rangos en el mapa de UG's.
- Marca las celdas seleccionadas del mapa.
- Puede haber celdas seleccionadas anteriormente y las respeta.

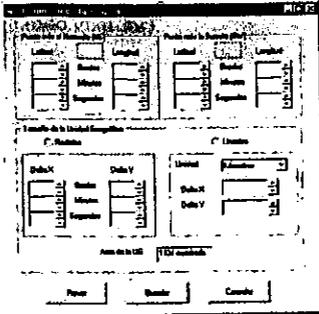
- **Paletas y rangos.**

- Cada análisis tiene definido un rango con un nombre.
- Hay asignación automática opcional de nombres, en caso de los intervalos constantes y percentiles.



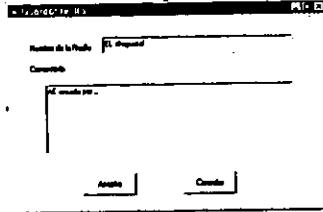
Edición de los intervalos porcentuales y absolutos.

• Definir rejilla.



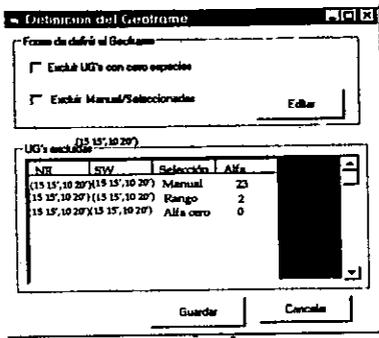
geoframe.

- Debe de haber una base de datos abierta.
- Se puede ver el mapa alfa y los análisis de localidades.
- Si se prevé, la manera de regresar es a través del menú y debe de recordar los valores pasados.



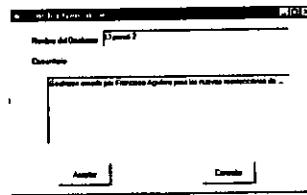
Interfaz para guardar rejilla (aparece también al seleccionar el comando "guardar" por primera vez)

- Definir



- Debe de haber una rejilla cargada o un geoframe base.
- Permite ver el mapa alfa.
- Se puede minimizar para selección manual o por rangos.
 - El cursor cambia a la forma de selección.
 - Se selecciona con un click y se deselecciona con otro.

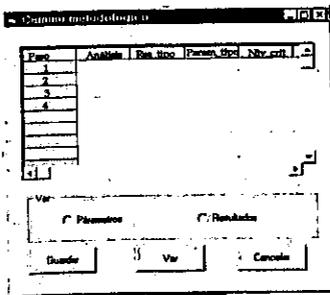
Las UG seleccionadas son excluidas y se resaltan



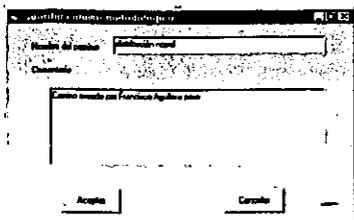
Interfaz de salvar Geoframe (aparece también si al editar el geoframe se selecciona el comando de "guardar" por primera vez. Se

puede indicar varias veces guardar mientras se define el geoframe, pero en el momento en que se pide un análisis diferente al mapa alfa y/o localidades, ese menú asume guardar camino metodológico).

• Camino Metodológico.

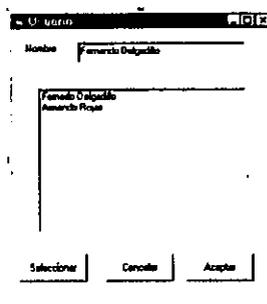


- Proporciona la descripción corta del camino.
 - Los parámetros numéricos ingresados.
 - Rango utilizado (nombre)
 - Tipo resultado.
 - Tipo parámetro.
 - Etc. (revisar BD).
- Permite ver cualquier paso del camino.
 - Tanto parámetros.
 - Resultados.
- En pasos anteriores no se permite ya la edición, no así en pasos siguientes.



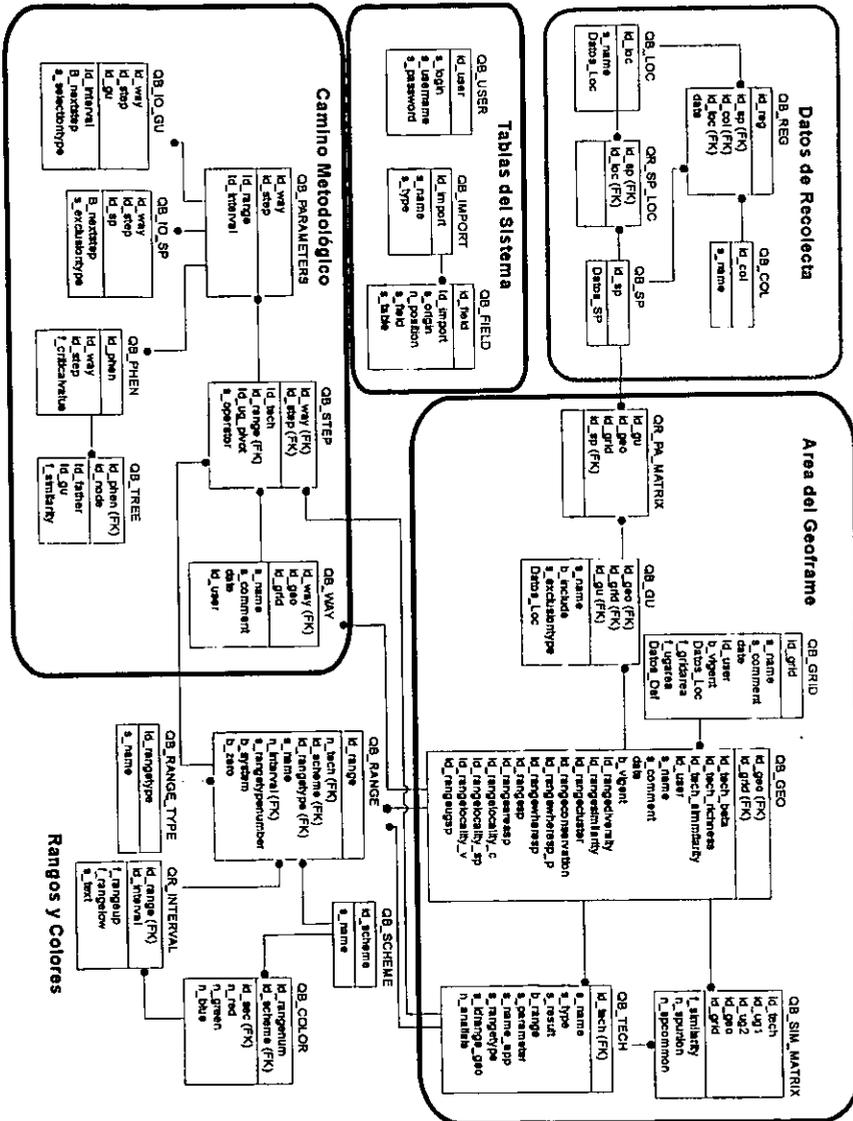
Interfaz de guardar camino (aparece por primera vez en el comando guardar)

- **Usuarios.**
 - Permitirá un control opcional de las modificaciones y trabajos que se han efectuado.



11 Apéndice 3 : Base de Datos (Diagrama ER y Diccionario de Datos.)

11.1 Diagrama ER



11.2 Diccionario de Datos:

- **Tabla QB_COL:** Contiene los nombres de los colectores.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_col	Número (entero)	2	Llave principal.
s_name	Texto	100	Nombre del colector.

- **Tabla QB_COLOR:** Contiene la definición de los colores

Columnas

Nombre	Tipo	Tamaño	Descripción
id_scheme	Número (entero)	2	Llave foránea de QB_SCHEMEC.
id_rangenum	Número (entero)	2	Número de rangos que soporta.
id_sec	Número (entero)	2	Secuencia del color en el esquema.
n_red	Número (Byte)	1	Componente roja en RGB.
n_green	Número (Byte)	1	Componente verde en RGB.
n_blue	Número (Byte)	1	Componente azul en RGB.

- **Tabla QB_FIELD :** Contiene la configuración de la importación de datos.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_field	Número (entero)	2	Llave primaria.
id_import	Número (entero)	2	Llave foránea de QB_IMPORT.
s_origin	Texto	50	Nombre del campo origen (1a. columna).
n_position	Número (entero)	2	Posición que ocupa el origen.
s_field	Texto	50	Campo destino.
s_table	Texto	50	Tabla destino.

- **Tabla QB_GEO :** Contiene el Geoframe y sus bioestadísticas.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_geo	Número (entero)	2	Llave principal.
id_grid	Número (entero)	2	Llave foránea de QB_GRID.
id_tech_beta	Número (entero)	2	Llave foránea de ID_TECH para la beta seleccionada.
id_tech_richness	Número (entero)	2	Llave foránea de ID_TECH para el criterio de estimación de riqueza.
id_tech_similarity	Número (entero)	2	Llave foránea de ID_TECH para el criterio de similitud.
id_user	Número (entero)	2	Llave foránea de QB_USER para el nombre del usuario.
s_name	Texto	50	Nombre del Geoframe.
s_comment	Memo	-	Comentario del geoframe.
date	Texto	10	Fecha de creación del geoframe.
b_vigent	Número (entero)	2	Si el geoframe sigue siendo vigente no ha habido actualizaciones de recolección. (-1 VERDADERO).
id_rangediversity	Número (entero)	2	Rango para el mapa alfa (Absoluto).
id_rangesimilarity	Número (entero)	2	Rango para el mapa de similitud (%).
id_rangecluster	Número (entero)	2	Rango para el mapa de fenograma (orden-agrupamiento).

id_rangeconservation	Número (entero)	2	Rango para el mapa de conservación (orden_agrupamiento).
id_rangewheresp_p	Número (entero)	2	Rango para el mapa de donde están las sp en OR (%)
id_rangewheresp	Número (entero)	2	Rango para el mapa de donde están las sp en AND (SI/NO/NA).
id_rangesp	Número (entero)	2	Rango para el mapa de cuales SP están n UG's (Absoluto).
id_rangeareasp	Número (entero)	2	Rango para el mapa de cuales UG's Contiene n SP (SI/NO/NA).
id_rangelocality_c	Número (entero)	2	Rango para el mapa de localidades colectores (Absoluto).
id_rangelocality_sp	Número (entero)	2	Rango para el mapa de localidades Especies (Absoluto).
id_rangelocality_v	Número (entero)	2	Rango para el mapa de localidades visitas (Absoluto).
id_rangeugsp	Número (largo)	4	Rango para el color de la lista de especies (and/or).

- Tabla QB_GRID : Contiene la definición del grid.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_grid	Número (entero)	2	Llave principal.
s_name	Texto	50	Nombre del Grid.
s_comment	Memo	-	Comentario del Grid.
date	Texto	10	Fecha de creación.
id_user	Número (entero)	2	Llave foránea de QB_USER para el nombre del usuario.
b_vigent	Número (entero)	2	Si el Grid sigue vigente (caso en que las nuevas recolectas amplíen geográficamente el AE). (-1 VERDADERO).
n_nwlat_degree	Número (entero)	2	Latitud del punto más al noreste en grados.
n_nwlat_minute	Número (entero)	2	Latitud del punto más al noreste en minutos
n_nwlat_second	Número (doble)	8	Latitud del punto más al noreste en segundos.
n_nwlon_degree	Número (entero)	2	Longitud del punto más al noreste en grados.
n_nwlon_minute	Número (entero)	2	Longitud del punto más al noreste en minutos.
n_nwlon_second	Número (doble)	8	Longitud del punto más al noreste en segundos.
n_selat_degree	Número (entero)	2	Latitud del punto más al suroeste en grados.
n_selat_minute	Número (entero)	2	Latitud del punto más al suroeste en minutos.
n_selat_second	Número (doble)	8	Latitud del punto más al suroeste en segundos.
n_selon_degree	Número (entero)	2	Longitud del punto más al suroeste en grados.
n_selon_minute	Número (entero)	2	Longitud del punto más al suroeste minutos.
n_selon_second	Número (doble)	8	Longitud del punto más al suroeste en segundos.
f_gridarea	Número (doble)	8	Área total.
f_ugarea	Número (doble)	8	Área de cada UG.
s_ugdefinition	Texto	50	Tipo de definición del grid (LINEAR para una definición en distancia y RADIAL para una definición en grados).
s_units	Texto	10	Tipo de unidades utilizadas en caso de definición lineal ("Km." kilómetros,"m" metros, "miles" millas).
f_deltax	Número (doble)	8	Delta X en caso de definición lineal.

f_deltay	Número (doble)	8	Delta Y en caso de definición lineal.
n_deltax_degree	Número (entero)	2	Delta X en caso de definición radial en grados.
n_deltax_minute	Número (entero)	2	Delta X en caso de definición radial en minutos.
n_deltax_second	Número (doble)	8	Delta X en caso de definición radial en segundos.
n_deltay_degree	Número (entero)	2	Delta Y en caso de definición radial en grados.
n_deltay_minute	Número (entero)	2	Delta Y en caso de definición radial en minutos.
n_deltay_second	Número (doble)	8	Delta Y en caso de definición radial en segundos.

- Tabla QB_GU : Contiene todas las GU's.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_geo	Número (entero)	2	Llave foránea de QB_GEO.
id_grid	Número (entero)	2	Llave foránea de QB_GRID.
id_gu	Número (entero)	2	Llave principal.
s_name	Texto	50	Nombre de la GU.
b_include	Número (entero)	2	Incluido en el Geoframe. (-1 VERDADERO).
s_exclusionstype	Texto	10	Tipo de exclusión de la UG (ZERO por alfa cero, MANUAL por selección manual).
n_nwlat_degree	Número (entero)	2	Latitud del punto más al noreste en grados.
n_nwlat_minute	Número (entero)	2	Latitud del punto más al noreste en minutos.
n_nwlat_second	Número (doble)	8	Latitud del punto más al noreste en segundos.
n_nwlon_degree	Número (entero)	2	Longitud del punto más al noreste en grados.
n_nwlon_minute	Número (entero)	2	Longitud del punto más al noreste en minutos.
n_nwlon_second	Número (doble)	8	Longitud del punto más al noreste en segundos.
n_selat_degree	Número (entero)	2	Latitud del punto más al sureste en grados.
n_selat_minute	Número (entero)	2	Latitud del punto más al sureste en minutos.
n_selat_second	Número (doble)	8	Latitud del punto más al sureste en segundos.
n_selon_degree	Número (entero)	2	Longitud del punto más al sureste en grados.
n_selon_minute	Número (entero)	2	Longitud del punto más al sureste en minutos.
n_selon_second	Número (doble)	8	Longitud del punto más al sureste en segundos.
n_loncenterpoint_degree	Número (entero)	2	Longitud del punto central en grados.
n_loncenterpoint_minute	Número (entero)	2	Longitud del punto central en minutos.
n_loncenterpoint_second	Número (doble)	8	Longitud del punto central en segundos.
n_latcenterpoint_degree	Número (entero)	2	Latitud del punto central en grados.
n_latcenterpoint_minute	Número (entero)	2	Latitud del punto central en minutos.
n_latcenterpoint_second	Número (doble)	8	Latitud del punto central en segundos.

- Tabla QB_IMPORT : Contiene los nombres de los formatos de importación.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_import	Número (entero)	2	Llave primaria.
s_name	Texto	50	Nombre de la importación.
s_type	Texto	10	RECORDS o SPECIES.

- Tabla QB_IO_GU :Contiene la entrada/salida de un paso del camino de GU.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_way	Número (entero)	2	Llave foránea de QB_WAY.
id_step	Número (entero)	2	Llave del paso.
id_gu	Número (entero)	2	Llave foránea de QB_GU.
id_interval	Número (entero)	2	Llave foránea de QR_INTERVAL que indica el intervalo en la que cayó la UG.
b_nextstep	Texto	1	Indica que en caso de ser resultado sí pasó como parámetro.
s_selectiontype	Texto	10	Tipo de selección de la UG, en caso de ser parámetro (MANUAL, ALL para todos, RANGE para caso de un rango).

- Tabla QB_IO_SP : Contiene la Entrada/Salida de un paso en el camino de SP.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_way	Número (entero)	2	Llave foránea de QB_WAY.
id_step	Número (entero)	2	Llave del paso.
id_sp	Número (entero)	2	Llave foránea de QB_SP.
b_nextstep	Texto	1	Indica si la especie paso como parámetro del siguiente análisis.
s_exclusiontype	Texto	10	Tipo de Exclusión (MANUAL; COMMON; UNION).

- Tabla QB_LOC : Contiene datos de las localidades.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_loc	Número (entero)	2	Llave principal.
s_name	Texto	255	Nombre de la Localidad.
n_lat_degree	Número (entero)	2	Latitud de la localidad en grados.0
n_lat_minute	Número (entero)	2	Latitud de la localidad en minutos.
n_lat_second	Número (doble)	8	Latitud de la localidad en segundos.
n_lon_degree	Número (entero)	2	Longitud de la localidad en grados.
n_lon_minute	Número (entero)	2	Longitud de la localidad en minutos.
n_lon_second	Número (doble)	8	Longitud de la localidad en segundos.

- Tabla QB_PARAMETERS : Contiene parámetros del paso (uno

Columnas

Nombre	Tipo	Tamaño	Descripción
id_way	Número (entero)	2	Llave foránea de QB_WAY.
id_step	Número (entero)	2	Llave foránea de QB_STEP.
id_range	Número (entero)	2	Llave foránea de QB_RANGE.
id_interval	Número (entero)	2	Llave foránea de QR_INTERVAL que indica los intervalos excluidos.

- Tabla QB_PHEN : Contiene los fenogramas del camino.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_phen	Número (entero)	2	Llave principal.
id_way	Número (entero)	2	Llave foránea de QB_WAY.
id_step	Número (entero)	2	Llave foránea de QB_STEP.
f_criticalvalue	Número (doble)	8	Valor crítico para conformar grupos.

- Tabla QB_RANGE : Contiene los rangos.

Columnas

Nombre	Tipo	Tamaño	Descripción
n_tech	Número (entero)	2	Llave foránea de QB_TECH.
id_scheme	Número (entero)	2	Llave foránea de QB_SCHEME.
id_rangetype	Número (entero)	2	Llave foránea de QB_RANGE_TYPE.
id_range	Número (entero)	2	Llave principal.
s_name	Texto	50	Nombre del rango.
n_interval	Número (entero)	2	Número de intervalos en color.
s_rangetypenumber	Texto	10	Indica el tipo de rango (PERCENT, ABSOLUTE).
b_system	Número (entero)	2	Indica si es del sistema el rango, de ser así no e modificable mas que salvado con otro nombre (-1 VERDADERO).
b_zero	Número (entero)	2	Indica si el rango contienen un intervalo cero (-1 VERDADERO).

- Tabla QB_RANGE_TYPE : Contiene los tipos de rangos.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_rangetype	Número (entero)	2	Llave Primaria.
s_name	Texto	20	Nombre del tipo de rango (MANUAL; PERCENTILES; EQUALI,...).

- Tabla QB_REG : Contiene los registros de recolección.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_reg	Número (entero)	2	Llave principal.
id_sp	Número (entero)	2	Llave foránea de QB_SP
id_col	Número (entero)	2	Llave foránea de QB_COL.
id_loc	Número (entero)	2	Llave foránea de QB_LOC.
date	Texto	10	Fecha de la recolección.

- Tabla QB_SCHEME : Contiene los esquemas de color.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_scheme	Número (entero)	2	Llave principal.
s_name	Texto	50	Nombre del esquema.

- Tabla QB_SIM_MATRIZ :Contiene la matriz de similitud para

Columnas

Nombre	Tipo	Tamaño	Descripción
id_tech	Número (entero)	2	Llave foránea de QB_TECH.
id_ug1	Número (entero)	2	Llave foránea de QB_GU.
id_ug2	Número (entero)	2	Llave foránea de QB_GU.
id_geo	Número (entero)	2	Llave foránea de QB_GEO.
id_grid	Número (entero)	2	Llave foránea de QB_GRID.
f_similarity	Número (doble)	8	Similitud entre los 2 UG's.
n_spunion	Número (entero)	2	Número de especies en total de las 2 UG's.
n_spcommon	Número (entero)	2	Número de especies en común de las 2 UG's.

- Tabla QB_SP : Nombres de las especies (y alguna categoría supraespecífica).

Columnas

Nombre	Tipo	Tamaño	Descripción
id_sp	Número (entero)	2	Llave principal.
s_supra	Texto	30	Nombre de categoría taxonómica supraespecífica .
s_genus	Texto	40	Nombre del género.
s_sp	Texto	40	Nombre de la especie.
s_author	Texto	50	Nombre del autor.

- Tabla QB_STEP: Contiene la secuencia de pasos de un camino.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_way	Número (entero)	2	Llave foránea de QB_WAY.
id_step	Número (entero)	2	Llave principal.
id_tech	Número (entero)	2	Llave foránea de QB_TECH.
id_range	Número (entero)	2	Llave foránea de QB_RANGE.
id_ug_pivot	Número (entero)	2	Llave foránea de QB_UG indicando cuál fue la UG Pivote en caso de un análisis de similitud.
s_operator	Texto	5	Tipo de operador (OR; AND).

- Tabla QB_TECH : Contiene la lista de los análisis y preguntas.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_tech	Número (entero)	2	Llave principal.
s_name	Texto	50	Nombre del análisis.
s_type	Texto	20	Nombre del grupo al que pertenece (DIVERSITY RICHNESS, SIMILARITY, QUALITY, CLUSTER, CONSERVATION, LOCALITY, WHERE SP, AREA SP, UG, SP, BETWEEN UG).
s_result	Texto	10	Tipo de resultado (UG SP, INTEGER, FLOAT).
b_range	Número (entero)	2	necesita rango.
s_parameter	Texto	10	Tipo de parámetro (UG, SP, INTEGER, FLOAT).

s_name_app	Texto	50	Nombre que aparece en Paletas y rangos.
s_rangetype	Texto	10	Tipo de rango [continuo absoluto= CONTINUO A continuo porcentual = CONTINUO P, si/no= SI/NO, and/or= AND/OR, n grupos= GRUPOS].
s_idrange_geo	Texto	20	Rango que modifica de QB_GEO el código.
n_analisis	Número (entero)	2	Para ligar con código si se puede o no.

- Tabla QB_TREE : Contiene los nodos del fenograma

Columnas

Nombre	Tipo	Tamaño	Descripción
id_phen	Número (entero)	2	Llave foránea de QB_PHEN.
id_node	Número (entero)	2	Llave del nodo.
id_father	Número (entero)	2	Llave del nodo padre.
id_gu	Número (entero)	2	Llave foránea de QB_GU.
f_similarity	Número (doble)	8	Grado de similitud.

- Tabla QB_USER :Nombres de usuarios de una instalación de QB-MAP.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_user	Número (entero)	2	Llave primaria.
s_login	Texto	15	Login.
s_username	Texto	50	Nombre del usuario.
s_password	Texto	10	Password encriptado.

- Tabla QB_WAY : Contiene los caminos metodológicos.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_way	Número (entero)	2	Llave principal.
id_geo	Número (entero)	2	Llave foránea de QB_GEO.
id_grid	Número (entero)	2	Llave foránea de QB_GRID.
s_name	Texto	50	Nombre del camino.
s_comment	Memo	-	Comentario del camino.
date	Texto	10	Fecha de creación (día/mes/año).
id_user	Número (largo)	4	Llave foránea de QB_USER para el nombre del usuario.

- Tabla QR_INTERVAL :Contiene los intervalos.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_range	Número (entero)	2	Llave foránea de QB_RANGE.
id_interval	Número (entero)	2	Secuencia del rango.
f_rangeup	Número (doble)	8	Límite superior (si es nulo indica el rango los menores que .
f_rangelow	Número (doble)	8	Límite inferior (si es nulo indica el rango los mayores que f_rangeup).
s_text	Texto	20	Texto a desplegar (v.g. "50 a 100"; "prioritaria").

- **Tabla QR_PA_MATRIZ:** Contiene la Matriz de Presencia/Ausencia, tabla de relación entre UG y SP.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_gu	Número (entero)	2	Llave foránea de QB_GU.
id_geo	Número (entero)	2	Llave foránea de QB_GEO.
id_grid	Número (entero)	2	Llave principal.
id_sp	Número (entero)	2	Llave foránea de QB_SP.

- **Tabla QR_SP_LOC:** Relación entre localidades y especies.

Columnas

Nombre	Tipo	Tamaño	Descripción
id_sp	Número (entero)	2	Llave foránea de QB_SP.
id_loc	Número (entero)	2	Llave foránea de QB_LOC.

12 Apéndice 4 : Ejemplo de Documentación - Clase UG-

12.1 Descripción General:

Es la Clase encargada de la Unidad Grafica. Es responsable de sus propiedades Biológicas (lógicas) y de despliegue (gráficas).

Utiliza la clase Cerror, para el control de errores.

12.2 Propiedades:

Seleccionable(P_valor As Boolean): Fija u obtiene si la UG es posible seleccionarla.

Seleccionada(P_valor As Boolean): Fija u obtiene si la UG esta seleccionada en estos momentos.

Grid(P_valor As Boolean): Fija si el grid (líneas exteriores) están visibles.

Identificador(P_ID As integer): Fija u obtiene el identificador de la UG Lógico.

NombreGL(P_ID As Long): Fija u obtiene el Nombre GL de la UG (Gráfico).

Letra(P_letra As Long): Fija la letra de la UG. El parámetro es debe ser obtenido de la creación de letras en OpenGL.

TextoOn(P_valor As Boolean): Fija si el texto contenido en la UG es visible.

Nombre(P_nombre As String): Fija u obtiene el nombre que se le asigno a la GU.

Alpha(P_valor As Integer): Fija u obtiene el valor alpha de la UG.

Especies(P_Especies As CLSP): Fija u obtienen la clase lista de especies que contienen la UG.

Geoframe(ByVal P_valor As Boolean): Fija u obtienen si la UG esta dentro del Geoframe.

12.3 Funciones y Métodos:

Control(P_cntrl As glxCtl): Le asigna ala clase el control OpenGL encargado de su despliegue.

FijaColor(P_colorRed As Integer, P_colorGreen As Integer, P_colorBlue As Integer): Fija el color de la UG de relleno.

Dibuja(): Llama a cada uno de los objetos de las capas para que se dibujen (a excepción de los rectángulos dinámicos hechos con el ratón). Debe ir en el evento Draw del control, o cada vez que queramos que el mapa se dibuje.

Posicion(P_X1 As Single, P_Y1 As Single, P_X2 As Single, P_Y2 As Single, Optional P_Cx As Single = 0, Optional P_Cy As Single = 0): Fija la posición de la UG, los parámetros opcionales son el centroide.

Exacta(P_nwlat_degree As Integer, P_nwlat_minute As Integer, P_nwlat_second As Double, P_nwlon_degree As Integer, P_nwlon_minute As Integer, P_nwlon_second As Double, P_selat_degree As Integer, P_selat_minute As Integer, P_selat_second As Integer, P_selon_degree As Integer, P_selon_minute As Integer, P_selon_second As Double, Optional ByVal P_regresa As Boolean = False): Asigna la posición exacta de la UG si el parámetro opcional es falso o por default, pero si el parámetro opcional es verdadero entonces regresa la posición exacta en vez de asignarla.

13 Glosario

Análisis multivariados: Conjunto de técnicas estadísticas para el análisis simultáneo de más de una variable independiente. Estos métodos incluyen técnicas de clasificación y ordenación, así como análisis de varianza y covarianza, regresión y correlación múltiple.

Biota: Conjunto de seres vivos de una región.

Catastro: Censo y padrón estadístico de las propiedades rústicas y urbanas de un país.

Endemismo : Especie u otra categoría taxonómica cualquiera, de planta o animal, cuya área de distribución es limitada y se circunscribe a una zona.

Filogenia: Especialidad de la biología que estudia la historia evolutiva de los distintos grupos de organismos e intenta reconstruir el proceso a base de los datos aportados por fósiles y la bioquímica.

Hábitat: Conjunto de condiciones ambientales, climáticas o biológicas que favorecen la vida y el desarrollo de una determinada especie.

Hibridación : Cruzamiento entre individuos diferenciados en más de un carácter hereditario. Artificialmente la hibridación se usa para la mejora de las razas de animales y planta.

Modelo Entidad Relación (ER): Es una herramienta para modelar bases de datos que describe las asociaciones que existen entre las diferentes categorías de datos dentro de un sistema de empresa o de información (no sólo dice cómo implantar, crear, usar o borrar datos).

Pseudocódigo: Es una herramienta que define los algoritmos detallados de los programas informáticos o la lógica de estos programas antes de la codificación de los mismos.

Bibliografía:

- Tesis

1. Lora Hernández, Esteban (1997) *"Diseño E Implementación De Un Sistema Para La Consulta De Usos De Suelo Con Una Base De Datos Multimodelo Y Un Lenguaje De 4ta. Generación."* Tesis inédita de Licenciatura. UNAM. México.
2. Rodríguez Castillo, Alejandro (1998) *"Desarrollo De Una Base De Datos Geográfica Y Estadística De La República Mexicana, De Utilidad En Aplicaciones De Sistemas Geográficos De Información."* Tesis inédita de Licenciatura. UNAM. México.

- Libros

1. Booch Grady (1991) *"Object Oriented Design with Applications"* Ed. The Benjamin/Cumming Publishing Company Inc.
2. Burrough P.A. (1996) *"Principle of Geographical Information systems for Land Resources Assessment"* Ed. Clareon Press Oxford.
3. Eddon, Guy. Eddon, Henry (1997) *"Active X Visual Basic 5.0"* Ed. McGraw-Hill.
4. Joly, Fernando. *"La cartografía."* Ed. Ariel, 1979.
5. Manual de SPANS (1993) *"SPANS GIS Learning System"* INTER TyDAC Technologies Inc 1993.
6. McConnell, Steve (1997) *"Desarrollo y Gestión de Proyectos Informáticos"* Ed. McGraw-Hill.
7. Stroupstrup, Bjarne (1993) *"El C++ lenguaje de Programación"* Ed. Addison-Wesley/Díaz de Santos.
8. Torge, Wolfgang (1983) *"Geodesia"* Ed. Diana Técnico.
9. Varios autores (1928) *"Geografía Universal, Descripción Moderna del Mundo, Tomo I, El espacio y la Tierra"* Publicación del Instituto Gallach de Librería y Ediciones, Barcelona.
10. Varios autores *"Biodiversidad, Taxonomía y Biogeografía de Artrópodos de México"* Editores Jorge Llorente Bousquets, Alfonso N García Aldrete, Enrique Gonzales Soriano.
11. Whitten, Jeffrey L. Bentley, lonnie D. Barlow, Víctor M (1997) *"Análisis y Diseño de Sistemas de Información"* Ed. McGraw-Hill/Irwin.
12. Wrigth Richard S, Sweet Michel (1996) *"Open GL Superbible"* Waite Group Press.

• **Revistas**

1. Dirzo, Rodolfo (1990) *"La Biodiversidad como Crisis Ecológica Actual ¿Qué Sabemos?"* Ciencias, Revista de Difusión, No. Especial, Julio.
2. Ezcurra, Ezequiel (1990) *"¿Por qué hay tantas especies raras?"* Ciencias, Revista de Difusión, No. Especial, Julio.
3. Krishock, Dan (1996) *"Servicios Públicos y GIS"* CADXPress, año I No 1-1996.
4. Mass, José Manuel. Martínez-Yrizar, Angelina (1990) *"Los Ecosistemas: definición, origen e importancia del concepto"* Ciencias, Revista de Difusión, No. Especial, Julio.
5. Nieto, José Ignacio (1998) *"En busca de Fahrenheit "* PC ACTUAL, Año IX, No. 95 Marzo.
6. Soberón M, Jorgue (1990) *"Ecotecnología, Predicción y Ciencia"* Ciencias, Revista de Difusión, No. Especial, Julio.
7. Zavala Hurtado, Alejandro. *"Introducción al Enfoque Multivariado en Estudios de Vegetación"*. Editor Gastón Editorial, Cuadernos de Divulgación INIREB, No 26.

• **Internet**

1. <http://www.ucmp.berkeley.edu/museum/datamodel/references.html>
2. **The Museum of Paleontology.**
University of California, Berkley.
3. <http://viceroy.eeb.uconn.edu/biota>
Specimen-Based, Biological Data and Collections Management Software for Individuals, Institutions, and Projects
Created by Robert K. Colwell, University of Connecticut. Published by Sinauer Associates
4. <http://www.baylor.edu/~grass/>
5. <http://www.lap.at/lap/laallgem/gis.html>
6. <http://www.gislinx.com/>
7. **Innovación en los Sistemas de Información Geográfica, Michel Gould**
<http://www.lander.es/~mgould/coloq-uab.html> (gould@inf.uji.es) .