



DIRECCIÓN DE PREVENCIÓN Y ATENCIÓN DE EMERGENCIAS

ACTUALIZACIÓN Y SISTEMATIZACIÓN DE LOS ESCENARIOS DE DAÑO POR TERREMOTO

INFORME FINAL PROTOTIPO DEL SISTEMA FASE I

**SUBDIRECCIÓN DE TÉCNICA Y DE GESTIÓN
COORDINACIÓN DE INVESTIGACIÓN Y DESARROLLO
GRUPO DE ESCENARIO DE DAÑOS**

31 DE AGOSTO DE 2009





DIRECCIÓN DE PREVENCIÓN Y ATENCIÓN DE EMERGENCIAS

GRUPO DE TRABAJO

DIRECCIÓN DE PREVENCIÓN Y ATENCIÓN DE EMERGENCIAS

GUILLERMO ESCOBAR CASTRO
Director

LUCY GONZÁLEZ
Responsable de la Subdirección de Técnica y de Gestión

GERMÁN BARRETO
Responsable de la Coordinación de Investigación y Desarrollo

Coordinación del proyecto
Junio 2008 a Febrero 2009

DULFAY ORTIZ
Responsable de Estudios Técnicos

Coordinación del proyecto
Marzo 2009 a Julio 2009

FERNANDO JAVIER DÍAZ PARRA
Responsable de Escenarios de Daños

Profesionales participantes
Junio 2008 a Junio 2009

FERNANDO DÍAZ
HAROLD ÁLVAREZ
ALEXYS RODRÍGUEZ
GERARDO RODRÍGUEZ
DIEGO TORRES



DIRECCIÓN DE PREVENCIÓN Y ATENCIÓN DE EMERGENCIAS

RESUMEN

La Coordinación de Investigación y Desarrollo de la Dirección de Prevención y Atención de Emergencias DPAE, tiene a cargo adelantar diferentes actividades enmarcadas en la estrategia del Plan de Respuesta por Terremoto, PRT, específicamente en los componentes de Escenarios de Daño, Sistema de Información y parte la información pública y conciencia ciudadana. De este modo, el presente documento resume los avances en la creación del software en su primera fase del plan de acciones de continuidad para actualizar y sistematizar los escenarios de daño por terremoto de la ciudad de Bogotá. El proyecto en mención se desarrolla desde febrero de 2008 a partir de la formulación del plan de acciones de continuidad e inicia actividades el mes de junio de 2008 con la contratación del personal requerido. El estudio tiene una duración la primera fase de 12 meses y la segunda fase de 12 meses.

El informe de ingeniería de software de la primera fase del proyecto esta organizado de acuerdo al desarrollo de las actividades, cubriendo los siguientes aspectos: en el capítulo dos se resumen los antecedentes, justificación, objetivos, alcances, el marco lógico de actividades y enfoque conceptual del proyecto; punto de partida para el capítulo tres que resume los objetivos específicos, alcance, requerimientos generales, plataforma tecnológica y arquitectura general del prototipo del sistema; en el capítulo cuatro se encuentra la metodología de desarrollo aplicada acorde con los procedimientos definidos por la Entidad para tal fin; en el capítulo cinco se esboza la descripción de las secciones y módulos que componen el prototipo del sistema a manera de manual de usuario; finalmente en el capítulo seis resume algunas de las pruebas y recomendaciones para la siguiente fase del proyecto.

CONTENIDO

1	INTRODUCCIÓN	1
2	ACTUALIZACIÓN Y SISTEMATIZACIÓN DE ESCENARIO DE DAÑOS POR TERREMOTO	2
2.1	GENERALIDADES DEL PROYECTO	2
2.1.1	ANTECEDENTES	2
2.1.2	JUSTIFICACIÓN	3
2.1.3	OBJETIVOS	4
2.1.4	ALCANCE	4
2.1.5	MARCO LÓGICO DE ACTIVIDADES	5
2.2	ENFOQUE CONCEPTUAL PARA EL DESARROLLO DEL PROYECTO	5
2.2.1	DESCRIPCIÓN DEL PROBLEMA	5
2.2.2	METODOLOGÍA DE EVALUACIÓN DEL RIESGO SÍSMICO PROPUESTA	7
3	GENERALIDADES DEL PROTOTIPO DEL SISTEMA	9
3.1	OBJETIVOS	9
3.2	ALCANCES	9
3.3	REQUERIMIENTOS	9
3.4	PLATAFORMA TECNOLÓGICA Y ARQUITECTURA	10
4	METODOLOGÍA DE DESARROLLO APLICADA – RUP	13
4.1	GENERALIDADES	13
4.1.1	CONCEPCIÓN	13
4.1.2	ELABORACIÓN	13
4.1.3	CONSTRUCCIÓN	14
4.1.4	TRANSICIÓN	15
4.2	DOCUMENTACIÓN DEL SISTEMA CON LOS FORMATOS ESTABLECIDOS POR LA ENTIDAD	15
4.2.1	LEVANTAMIENTO DE REQUISITOS FUNCIONALES	15
4.2.2	ANÁLISIS DE REQUISITOS FUNCIONALES	15
4.2.3	CASOS DE USO	15
4.2.4	ANÁLISIS DE CASOS DE USO	15
4.2.5	MODELO DE DATOS	16
4.2.6	DICCIONARIO DE DATOS	19
4.2.7	DESPLIEGUE DE LA APLICACIÓN	19
5	DESCRIPCIÓN DEL PROTOTIPO DEL SISTEMA	21
5.1	REQUERIMIENTOS DE SOFTWARE	21



5.2	LANZAMIENTO DE LA APLICACIÓN.....	21
5.3	PANTALLA PRINCIPAL.....	23
5.4	SECCIONES	23
6	<u>REALIZACIÓN DE PRUEBAS Y RECOMENDACIONES</u>	<u>28</u>
6.1	PROCEDIMIENTOS MEJORADOS DURANTE EL DESARROLLO	28
6.2	RECOMENDACIONES PARA LA SEGUNDA FASE DEL PROYECTO.....	30
7	<u>CONCLUSIONES Y RECOMENDACIONES.....</u>	<u>35</u>
8	<u>REFERENCIAS</u>	<u>36</u>

LISTA DE FIGURAS

Figura 2.1. Definición del problema.....	6
Figura 2.2. Cadena de acontecimientos que definen el problema.....	6
Figura 2.3. Tipos de análisis que se desarrollan para afrontar el problema	7
Figura 2.4. Interacción de las componentes del análisis de riesgo sísmico	8
Figura 3.1. Diagrama de procesos para el cálculo de escenario de daños	10
Figura 3.2. Arquitectura general y relación con SIG del FOPAE	11
Figura 3.3. Arquitectura de la aplicación	12
Figura 4.1. Esfuerzo en actividades según la fase del proyecto (Fuente: IBM).....	13
Figura 4.2. Esquema de trabajo del modelo de datos espaciales	16
Figura 4.3. Proceso de configuración de métodos	18
Figura 4.4. Árbol de procesos de un escenario de riesgo	18
Figura 4.5. Base de datos geográfica del sistema de cálculo según el modelo de datos.....	19
Figura 5.1. Pantalla "Abrir con".	22
Figura 5.2. Verificación de la versión de Java en la consola de DOS	22
Figura 5.3. Pantalla Principal	23
Figura 5.4. Menú de Opciones del Módulo de Registro	24
Figura 5.5. Menú de Opciones del Módulo de Daños, Afectaciones y Pérdidas	25
Figura 5.6. Menú de Opciones de Evaluación Global del Riesgo.....	26
Figura 5.7. Menú de opciones del módulo de Resultados.....	27
Figura 5.8. Menú Ayuda de la Aplicación	27
Figura 6.1. Esquema de registro anterior	29
Figura 6.2. Esquema de registro actual.....	29
Figura 6.3. Resultados preliminares del prototipo de índice de vulnerabilidad de edificaciones	31

Figura 6.4. Resultados preliminares del prototipo de índice de daño de edificaciones	32
Figura 6.5. Resultados preliminares del prototipo de índice de vulnerabilidad de la red de acueducto matriz	33
Figura 6.6. Resultados preliminares del prototipo de tasa de roturas de la red de acueducto matriz	34

1 INTRODUCCIÓN

La Coordinación de Investigación y Desarrollo de la Dirección de Prevención y Atención de Emergencias DPAE, tiene a cargo adelantar diferentes actividades enmarcadas en la estrategia del Plan de Respuesta por Terremoto, PRT, específicamente en los componentes de Escenarios de Daño, Sistema de Información y parte la información pública y conciencia ciudadana. Para el tema de Escenario de Daño el grupo de Estudios Técnicos formuló un plan de acciones de continuidad para actualizar y sistematizar los escenarios de daño por terremoto de la ciudad de Bogotá.

El plan de acciones comprende en general el estudio de nuevas metodologías, modelos y funciones relacionadas con el cálculo de la amenaza, vulnerabilidad y riesgo sísmico, incorporando en los análisis efectos colaterales como deslizamientos y licuación de terrenos. Se recopilará la información disponible necesaria para evaluar los modelos escogidos y se organizará esta en un sistema de información geográfica y se desarrollará un software para el cálculo de los escenarios de daño por terremoto que sea posible de mejorar sus algoritmos y actualizar posteriormente con nueva información.

El proyecto en mención se desarrolla desde febrero de 2008 a partir de la formulación del plan de acciones de continuidad e inicia actividades el mes de junio de 2008 con la contratación del personal requerido. El estudio tiene una duración la primera fase de 12 meses y la segunda fase de 12 meses.

De este modo, el informe de ingeniería de software de la primera fase del proyecto esta organizado de acuerdo al desarrollo de las actividades, cubriendo los siguientes aspectos: en el capítulo dos se resumen los antecedentes, justificación, objetivos, alcances, el marco lógico de actividades y enfoque conceptual del proyecto; punto de partida para el capítulo tres que resume los objetivos específicos, alcance, requerimientos generales, plataforma tecnológica y arquitectura general del prototipo del sistema; en el capítulo cuatro se encuentra la metodología de desarrollo aplicada acorde con los procedimientos definidos por la Entidad para tal fin; en el capítulo cinco se esboza la descripción de las secciones y módulos que componen el prototipo del sistema a manera de manual de usuario; finalmente en el capítulo seis resume algunas de las pruebas y recomendaciones para la siguiente fase del proyecto.

2 ACTUALIZACIÓN Y SISTEMATIZACIÓN DE ESCENARIO DE DAÑOS POR TERREMOTO

Con el fin de contextualizar el informe de ingeniería de software a continuación se resumen los principales aspectos del proyecto de actualización y sistematización de escenarios de daño por terremoto, el detalle de la información y métodos desarrollados puede ser consultado en el informe técnico de la primera fase del proyecto.

2.1 GENERALIDADES DEL PROYECTO

2.1.1 Antecedentes

La ciudad de Bogotá, Distrito Capital de Colombia, de acuerdo con los estudios geológicos de la región está localizada en las inmediaciones de un ambiente sismotectónico que presente una actividad sísmica importante. Aunque en los últimos años no ha ocurrido un sismo intenso por la baja frecuencia relativa de eventos fuertes, no es correcto suponer que así han de seguir las cosas, por lo que evaluar las condiciones de riesgo sísmico en la ciudad es prioritario.

La ciudad de Bogotá de acuerdo con el Estudio General de Amenaza Sísmica de Colombia (1996), se localiza en una zona de amenaza sísmica intermedia, con una aceleración máxima probable A_a de 0.2 g para un periodo de retorno de 475 años. Posteriormente, en el Estudio de Microzonificación Sísmica de Bogotá (1997), se identificaron las principales fuentes sismogénicas que dominan la amenaza sísmica en la ciudad, obteniendo así que la Falla Frontal de los Llanos Orientales puede generar un evento de magnitud 7.2 a una distancia de 60 km, las fallas cercanas a la ciudad con una distancia menor de 20 km pueden llegar a generar eventos con magnitud de 6.0, y debido a la presencia de suelos blandos de espesores considerables en el norte de la ciudad, se considero la fuente lejana de la zona de subducción del pacífico a 400 km de distancia que puede generar eventos de magnitud superior 8.0.

Por otro lado, la ciudad presenta un desarrollo heterogéneo de sus construcciones, debido principalmente a que el primer código de construcción sismorresistente fue del año 1984, el cual fue posteriormente actualizado en el año 1998. Además, las condiciones socioeconómicas de la mayoría de los habitantes de la ciudad ha llevado a que las viviendas no cuenten con un diseño sismorresistente adecuado. Teniendo en cuenta que en la ciudad se encuentra las principales entidades del gobierno nacional, regional y distrital, así como las industrias de producción, bienes y servicios más importantes del país, la posible afectación en estas puede llegar a causar un gran impacto en el desarrollo del país. Con base en el panorama anterior la ciudad de Bogotá presenta un alto grado de vulnerabilidad sísmica, entendido como una baja capacidad de resistir por sus características estos elementos la amenaza que en este caso corresponden a las acciones de un sismo de considerable energía.

Teniendo en cuenta lo anterior, es necesario plantear escenarios de daño, que consisten básicamente en proponer panoramas hipotéticos, de acuerdo con los eventos sísmicos probables, que presenten áreas afectadas según las características de las construcciones, número de afectados (heridas y víctimas) diferenciadas si el evento ocurre de día o de noche y puntos críticos de redes de abastecimiento de servicios públicos, entre otros. Este tipo de escenarios sirve entre otros para propósitos de mitigación de riesgo sísmico, toma de consciencia, planificación para la atención de emergencias originadas por terremotos y planes de rehabilitación y reconstrucción.

Esta problemática ha llevado a que se adelanten desde hace varios años diversas actividades en torno a la gestión general del riesgo sísmico del área urbana de la ciudad. Con anterioridad a este programa se han realizado estudios que han involucrado temas aparentemente similares como el Estudio de Microzonificación Sísmica de Bogotá (1997), a través de un convenio interadministrativo entre la Unidad para la Prevención y Atención de Emergencias UPES (actual DPAE), el Instituto

Colombiano de Geología y Minería INGEOMINAS y la Universidad de los Andes UNIANDES; posteriormente, la Agencia de Cooperación Internacional del Japón JICA realizó el Estudio para la Prevención de Desastres en el Área Metropolitana de Bogotá (2002), y durante el año 2005 por medio de un convenio interadministrativo entre la DPAAE y la Universidad de los Andes, se elaboró el Estudio de Escenarios de Riesgo y Pérdida por Terremoto para Bogotá (2005).

En general, cada uno de los estudios relacionados anteriormente presenta un panorama del riesgo sísmico de la ciudad, los cuales han servido para promover los planes de prevención y atención de emergencias en la ciudad. Con base en ellos se está trabajando en la actualidad en el Plan de Respuesta por Terremoto, PRT, con las entidades del Sistema Distrital de Prevención y Atención de Emergencias, SDPAE, el cual pretende fortalecer a las entidades del sistema en los temas de prevención y atención en el caso de ocurrir un terremoto.

2.1.2 Justificación

Tras de varios años trabajando en el tema de la gestión del riesgo sísmico, se han detectado una serie de necesidades y aspectos que se deben contar en los estudios de escenarios de daño por terremoto, dentro de estos aspectos están los siguientes:

- Debido al rápido desarrollo de la ciudad y el conocimiento de nuevos elementos sobre amenaza, vulnerabilidad y riesgo, es necesario establecer escenarios de daños acordes con la Bogotá actual, garantizándose de alguna manera que como procesos dinámicos sea producto de actualización a lo largo del tiempo según el eventual cambio de las condiciones en el Distrito Capital, ya sean estos originados por la modificación en la vulnerabilidad de la infraestructura, la población o sencillamente por la expansión del perímetro urbano entre otros factores.
- La reducción de la vulnerabilidad sísmica se hace evidente a través de los programas de reforzamiento de puentes vehiculares y peatonales (rutas de Transmilenio por ejemplo), edificaciones indispensables, hospitales, escuelas, entre otras. Así mismo el crecimiento en áreas subnormales de la ciudad y la continuidad de la construcción informal, imponen mayores vulnerabilidades. De allí la necesidad de contar con un sistema dinámico de cálculo de escenarios de daños.
- Es necesario mejorar los algoritmos empleados indirectamente para identificar con la información de la base de datos catastral, las características estructurales de las edificaciones, indispensables para estimar su vulnerabilidad sísmica, teniendo en cuenta que en el estudio de Uniandes de 2005 uno de los aspectos a mejorar es la tipificación de los predios sin edificar, dotacionales e institucionales.
- El estudio de Uniandes de 2005 no contempló efectos colaterales o cosísmicos, tales como deslizamientos, licuación de terrenos, fallas de represas o jarillones que desencadenen posibles inundaciones y la ocurrencia de incendios por fallas en las redes de suministro de gas domiciliario o en el sector industrial. Los cuales dependiendo de la magnitud del evento pueden llegar a generar daños importantes.
- La actualización de la Microzonificación Sísmica de Bogotá mejora considerablemente la definición de las zonas de comportamiento sísmico homogéneo, por lo cual es necesario ajustar el escenario de daños a los nuevos valores de amenaza y zonas de comportamiento sísmico del escenario de daños.
- La implementación de la telemetría en la Red de Acelerógrafos de Bogotá, RAB, puede llegar a identificar en un breve tiempo las características de un movimiento fuerte sentido en la ciudad, lo cual puede articularse sistemáticamente para la definición aproximada del escenario de daño producido por el evento.
- El Plan de Respuesta por Terremoto, PRT, y las entidades del Sistema Distrital de Prevención y Atención de Emergencias, SDPAE, requieren de un escenario de daños por terremoto que represente las condiciones actuales en que está la ciudad.

Todos los anteriores aspectos llevan a la necesidad de adelantar una serie de trabajos y actividades tendientes a actualizar los estudios de riesgo sísmico de la ciudad.

2.1.3 Objetivos

El objetivo general del estudio es el diseñar e implementar un sistema dinámico de cálculo de escenario de daños por la ocurrencia de un sismo de gran magnitud, para los diferentes elementos expuestos.

Dado que el proyecto comprende un manejo de un volumen considerable de información se plantean dos fases para el desarrollo del objetivo general, de este modo se presentan los objetivos específicos de la Fase I a continuación.

- Definición del modelo de amenaza sísmica: revisar y plantear un modelo general de amenaza sísmica y comportamiento dinámico de rocas y suelos con base en la actualización de la microzonificación sísmica de la ciudad, con el fin de evaluar la variación de la respuesta dinámica para diferentes periodos de retorno por cada una de las fuentes sísmicas y estimar en superficie valores de aceleración, velocidad y desplazamiento espectral.
- Definición de modelos de vulnerabilidad: revisar y proponer modelos generales para estimar la vulnerabilidad sísmica de edificaciones, infraestructura, redes, y población en función de la amenaza sísmica en superficie.
- Definición de funciones de afectación y pérdidas económicas por sismo: revisar y proponer funciones para evaluar las afectaciones y pérdidas económicas por sismo de edificaciones, infraestructura, redes, y población, con el fin de contar con un orden de magnitud de los daños.
- Recopilación de información: con base en los modelos estudiados y los parámetros necesarios para evaluar la amenaza, vulnerabilidad y riesgo sísmico en la ciudad, se procede a recopilar y organizar en bases de datos y en el sistema de información geográfica la información disponible, tal como las características geotécnica de la ciudad, las zonas de respuesta sísmica, base de datos catastral de predios, censo de población, edificaciones indispensables y líneas vitales.
- Sistematización de procesos: diseñar e implementar un sistema de cálculo de tipo modular en versión prototipo para la plataforma tecnológica de la entidad para estimar escenarios de daños, que se pueda actualizar posteriormente cada uno de los módulos del sistema.
- Modelación de escenarios sísmicos de daño: a partir de los escenarios sísmicos definidos para el análisis, la información disponible, los modelos de vulnerabilidad, funciones de afectación y pérdida para las distintas temáticas trabajadas, se evaluará mediante el software desarrollado las afectaciones en términos de daños para edificaciones, infraestructura y redes; en número de afectados, heridos y víctimas, y en millones de pesos para las posibles pérdidas económicas, esto se efectuó de manera preliminar para realizar pruebas y calibrar los modelos.

2.1.4 Alcance

El alcance del proyecto está relacionado directamente con la información disponible para la primera fase, además el nivel de detalle que se logre dependerá de varios aspectos como: la cantidad y calidad de los datos, la escala de trabajo, las unidades de análisis definidas (predio, manzana, sector, etc.) y las limitaciones propias de los modelos empleados, entre otros.

El software desarrollado pretenderá ilustrar la magnitud de los posibles daños por un terremoto para una fuente sísmica y probabilidad de ocurrencia determinada, abarcando en la medida de lo posible todo el sector urbano y rural de la ciudad que cuente con información suficiente para aplicar los modelos.

Los resultados del proyecto no reemplazan los análisis de vulnerabilidad sísmica y estudios detallados para evaluar el riesgo sísmico de estructuras indispensables y líneas vitales. Sin embargo, los resultados servirán para identificar los sectores en mayor riesgo de modo que las entidades a cargo de la infraestructura indispensable de la ciudad puedan priorizar sus planes de mejoramiento y adecuación sísmo-resistente.

2.1.5 Marco lógico de actividades

La primera fase del proyecto se desarrolló siguiendo un marco lógico de actividades acorde con las necesidades planteadas en la propuesta de trabajo, luego para abordar los objetivos del estudio se plantea cinco actividades generales como se resumen a continuación.

- **Revisión de estudios y metodologías:** un primer paso para desarrollar el proyecto consiste en la revisión de los principales antecedentes relacionados con el objetivo del proyecto, esta actividad demanda de una exhaustiva investigación de metodologías desarrolladas para afrontar los diferentes temas del proyecto, con ello se logra definir con mayor certeza el alcance del proyecto, así como la información necesaria para el desarrollo de los métodos seleccionados para ser aplicados.
- **Recopilación de información:** posterior a la revisión de las metodologías que se pueden aplicar para el proyecto se realiza la recopilación de la información necesaria conforme los métodos seleccionados, esta actividad demanda además de la gestión para conseguir la información, la organización de misma y el procesamiento primario para su posterior empleo.
- **Desarrollo de los métodos para la evaluación de daños, afectaciones y pérdidas:** una vez se cuente con una visión clara de las metodologías aplicables y la información disponible, se procede a desarrollar, adaptar o adoptar los métodos detallados en forma de algoritmo de cálculo paso a paso para que puedan ser implementados o sistematizados.
- **Desarrollo del prototipo del sistema de cálculo:** tras contar con los requisitos funcionales de cada uno de los algoritmos de los métodos seleccionados para el proyecto se evalúan las diferentes opciones para desarrollar el sistema teniendo en cuenta la plataforma tecnológica de la entidad, la información que se maneja y bases de datos, los métodos que se quieren implementar y la disponibilidad de software.
- **Análisis de resultados de las pruebas y recomendaciones para la siguiente fase del proyecto:** se analizan los resultados de las pruebas obtenidas de los métodos implementados en el prototipo del sistema con el fin de evidenciar aspectos que se pueden mejorar en los procesos de cálculo, los algoritmos, las interfaces, entre otros, para luego dar recomendaciones y realizar propuestas de mejoramiento para la siguiente fase del proyecto.

2.2 ENFOQUE CONCEPTUAL PARA EL DESARROLLO DEL PROYECTO

Se plantea el enfoque conceptual con el cual se va desarrollar el proyecto, el cual ha sufrido algunos cambios con respecto a la versión inicialmente planteada en el plan de acciones de continuidad, dado que a la luz del conocimiento adquirido en las primeras etapas del proyecto, permitió mejorar dicho enfoque conceptual.

2.2.1 Descripción del problema

La ocurrencia de un evento sísmico desencadena una serie de acontecimientos o eventos que se ilustran de manera esquemática en la **Figura 2.1**. Cuando las ondas sísmicas llegan a un centro poblado los elementos que la componen experimentan o responden de una manera particular a las ondas, esta respuesta puede llegar a ocasionar daños cuando supera su capacidad, una vez ocurren los daños iniciales por efecto directo de las ondas estos pueden ocasionar efectos secundarios que se pueden llegar a propagar y producir mas daños, esta secuencia de acontecimientos se resumen en la **Figura 2.2**, los tipos de análisis que se realizan se pueden llegar a clasificar en análisis de las amenazas, vulnerabilidad y riesgo como se muestra en la **Figura 2.3**.

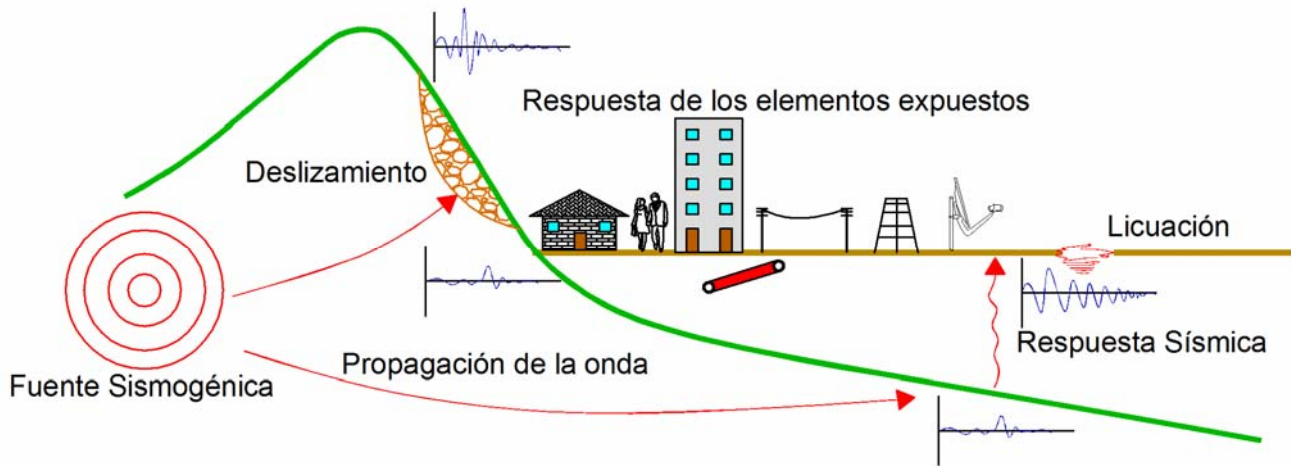


Figura 2.1. Definición del problema

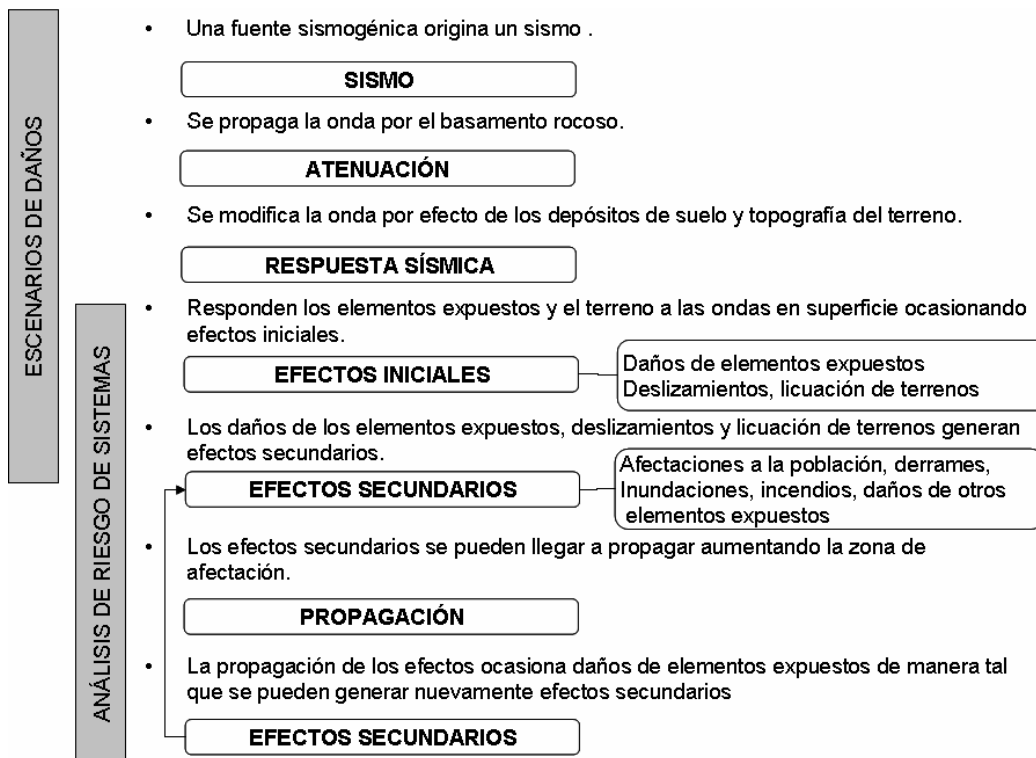


Figura 2.2. Cadena de acontecimientos que definen el problema

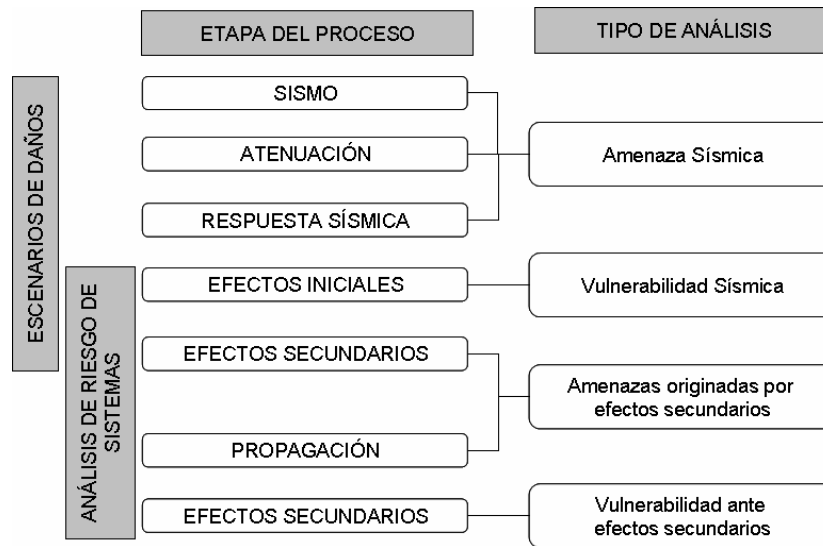


Figura 2.3. Tipos de análisis que se desarrollan para afrontar el problema

Es importante destacar que los resultados de riesgo en términos de daños y afectaciones del primer análisis de amenaza y vulnerabilidad, para el siguiente análisis se convierten en las amenazas, es decir que las consecuencias iniciales de un terremoto son detonantes o las causas de otros análisis de riesgo. A manera de ejemplo, para ilustrar la situación, imaginémos que ocurre un sismo en la falla frontal (sismo), se propaga la onda sísmica por el basamento rocoso (atenuación), por efecto de los suelos y topografía se amplifica (respuesta sísmica), las ondas en superficie mueven los edificios y le ocasionan daños a las estructuras (repuesta de los elementos – efectos iniciales), el daño de los edificios se convierte en una amenaza potencial para sus residentes y entorno, si llega a colapsar alguna estructura afectaría una zona de impacto (propagación) produciendo daños a otros edificios, redes, afectados y víctimas (efectos secundarios), los efectos secundarios se pueden propagar y volver a generar daños, los cual se puede repetir varias veces.

El anterior panorama describe de manera general los efectos directos que ocasiona un terremoto, adicional a toda la cadena de acontecimientos hay implicaciones indirectas en la sociedad, economía y el desarrollo de la ciudad, las cuales son aún más difíciles de evaluar. Para fines del proyecto de escenarios de daños es imprescindible ilustrar bajo este enfoque el alcance de las evaluaciones que realiza el estudio, las cuales se concentran en la cuantificación de los efectos iniciales y algunos secundarios (ver Figura 2.2). Para abarcar las evaluaciones de efectos secundarios es necesario estudiar varios tipos de detonantes (amenazas) por ejemplo: colapso de estructuras, derrames de aguas residuales o líquidos peligrosos, incendios, etc., la manera que se propagan y la vulnerabilidad de los elementos expuestos frente a este tipo de amenazas, siendo entonces un problema aún más complejo que se enmarca en los estudios de análisis de riesgo de sistemas urbanos.

2.2.2 Metodología de evaluación del riesgo sísmico propuesta

Los análisis de riesgo sísmico a nivel mundial se realizan desde hace ya varias décadas, las diferentes metodologías desarrolladas identifican diferentes componentes que entran a ser valorados en dichas evaluaciones. En general se ha llegado a un consenso para estimar el riesgo sísmico a partir de tres componentes básicas que son la amenaza, la vulnerabilidad y la capacidad de respuesta, de este modo el riesgo sísmico se define a partir de la siguiente expresión.

$$R \approx \frac{A \cdot V}{CR}$$

La amenaza (A) se define usualmente en términos de la probabilidad de ocurrencia, sin embargo también se puede determinar de manera determinística para una fuente sísmica particular, el valor de la amenaza será función principalmente del ambiente sismotectónico de la zona y las condiciones locales de los suelos. Además de los efectos directos sobre una

ciudad por la propagación de las ondas de un sismo, éste puede desencadenar otros efectos como deslizamientos, licuación de terrenos, inundaciones, etc., o a consecuencia de los daños de las estructuras pueden ocurrir incendios, de tal manera que la amenaza debería involucrar tanto efectos directos como indirectos (colaterales).

La vulnerabilidad (V) de los elementos expuestos (edificaciones, líneas vitales, población) representa la capacidad del elemento para resistir daño o afectación frente a la amenaza, es decir que un elemento con vulnerabilidad baja es capaz de resistir altos niveles de amenaza, en cambio un elemento con vulnerabilidad alta tiene poca resistencia para absorber la demanda de la amenaza, por lo que puede sufrir daños fácilmente.

Finalmente, la capacidad de respuesta (CR) para efectos del presente estudio hace referencia a la capacidad de la sociedad, instituciones y población, de reaccionar adecuadamente en los instantes de emergencia y responder al impacto, controlarlo y recuperarse, es decir que una sociedad con alta capacidad de respuesta reflejado en su buena preparación, organización, recursos para atender una emergencia, capacidad económica y de gestión para recuperarse, sufrirá un menor impacto que una sociedad con baja capacidad de respuesta, la cual por su falta de preparación, organización, recursos para la emergencia y para recuperarse hacen que el riesgo pueda aumentar.

Las estrategias para reducir el riesgo sísmico bajo el anterior enfoque se pueden agrupar en: iniciativas para la reducción de la vulnerabilidad de los elementos expuestos y gestión para el mejoramiento de la capacidad de respuesta de la sociedad.

El proyecto de actualización y sistematización de escenarios de daño por terremoto en su primera fase abordó las componentes de amenaza y vulnerabilidad. La segunda fase desarrollará la componente de capacidad de respuesta.

En la **Figura 2.4** se ilustra los diferentes componentes y elementos expuestos tenidos en cuenta para las evaluaciones de daño y posterior riesgo sísmico. La amenaza considera tanto la acción de las ondas sísmicas (respuesta sísmica) como los deslizamientos y licuación de terrenos (efectos colaterales); la vulnerabilidad se evalúa para edificaciones - normales e indispensables y líneas vitales – redes y estructuras esenciales; al conjugar la amenaza y la vulnerabilidad se obtienen indicadores de daño, afectaciones a la población y pérdidas económicas; la capacidad de respuesta de la sociedad, instituciones y población, se evalúa por medio de indicadores.



Figura 2.4. Interacción de las componentes del análisis de riesgo sísmico

Finalmente, se escoge una unidad de análisis de interés para integrar los resultados e indicadores mediante técnicas de evaluación multicriterio para obtener un índice de riesgo sísmico, el cual se puede interpretar más fácil que los resultados dispersos de las evaluaciones puntuales, este índice sirve para orientar a los tomadores de decisiones en la definición y/o mejora de las estrategias de reducción del riesgo sísmico, como por ejemplo los planes de respuesta por terremoto.

3 GENERALIDADES DEL PROTOTIPO DEL SISTEMA

Con base en el enfoque conceptual del proyecto presentado en el numeral 2.2, los requerimientos generales de cada uno de los métodos seleccionados para evaluar los posibles daños, afectaciones y pérdidas de los diferentes elementos expuestos considerados, la cantidad y calidad de la información disponible para aplicar dichos métodos, se precisa a continuación los objetivos específicos y alcance del prototipo del sistema resultado de la primera fase del proyecto.

3.1 OBJETIVOS

- Diseñar e implementar un prototipo de programa que permita probar los modelos revisados en la primera fase del proyecto de daño, afectación y pérdidas dadas unas condiciones de amenaza sísmica y elementos expuestos que estén organizados acorde a los requerimientos de los modelos implementados.
- Diseñar un modelo de datos espacial acorde con el enfoque general del proyecto y los requerimientos específicos de cada método, el cual sirva para la administración de los datos y la creación de la meta data de los resultados.
- Con base en los resultados de las pruebas realizadas con el prototipo se procederá a validar y calibrar los algoritmos que se desarrollen, así como analizar la eficiencia, duración y precisión de los procesos.

3.2 ALCANCES

- El prototipo del sistema solamente realizará los procesos de los métodos implementados partiendo de la información previamente preparada para su aplicación y los parámetros que el método requiera definidos por el grupo.
- La articulación completa entre el modelo de datos espacial y los diferentes módulos programados del prototipo no es objeto de esta fase, los resultados quedaran almacenados en las coberturas de trabajo.
- Dado que el objetivo principal es probar los métodos seleccionados, la cantidad y calidad de información disponible, el modelo de datos espacial, entre otros, los resultados obtenidos de las corridas de prueba no son los definitivos, únicamente sirven para detectar fallos del sistema y orientar la fase posterior del desarrollo.
- El prototipo del sistema cuenta con las interfaces básicas para el lanzamiento de las rutinas de los diferentes módulos, no es objeto de esta fase presentar el diseño completo de las interfaces, lo cual esta según el plan de acción de proyecto programado para la segunda fase del proyecto.

3.3 REQUERIMIENTOS

- De acuerdo con los objetivos y alcance dados para definir la plataforma y arquitectura del prototipo se requiere basar su desarrollo en una herramienta de Sistemas de Información Geográfica con las siguientes características fundamentales:
- Compatibilidad con las tecnologías SIG e información geográfica disponibles en el FOPAE.
- Capacidad de procesar grandes volúmenes de información geográfica en diferentes formatos, y en diferentes fuentes de datos espaciales.

- Eficiencia en las corridas de los escenarios de daño, en términos de velocidad de procesamiento. Es por esta razón que se sugiere el uso de aplicativos de escritorio con mínima utilización y acceso a recursos de red tanto institucional como de otras redes externas.
- Poderoso ambiente de desarrollo, dada la complejidad y la especificidad de las labores de geoprocésamiento de los escenarios de daño; se requiere el uso frecuente de operaciones y funciones matemáticas especializadas, y la facilidad de utilización de los objetos básicos del software SIG para la manipulación personalizada de datos espaciales tanto en formato vectorial como raster.

En la **Figura 3.1** se ilustra de manera esquemática los tipos de procesos que se requieren desarrollar.

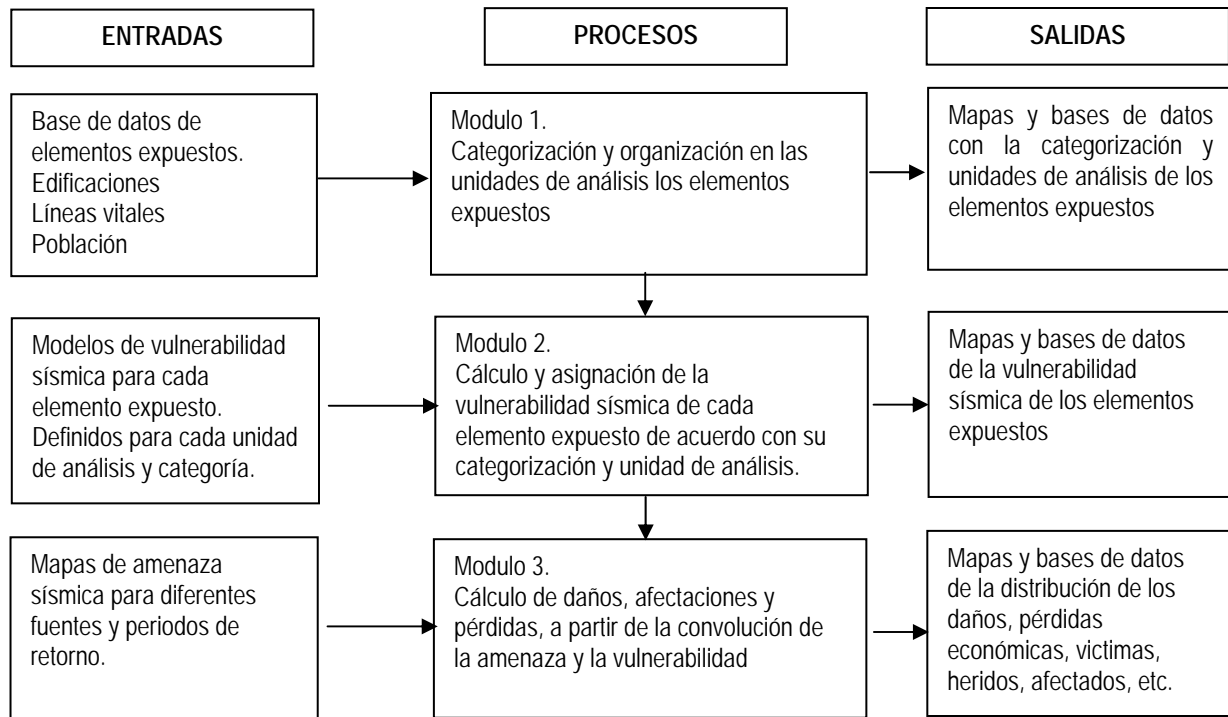


Figura 3.1. Diagrama de procesos para el cálculo de escenario de daños

3.4 PLATAFORMA TECNOLÓGICA Y ARQUITECTURA

La arquitectura general y su relación con SIG del FOPAE se muestra en la **Figura 3.2**, donde a nivel de usuario el software es un aplicativo de escritorio "stand alone", que consiste en una herramienta con varias funcionalidades en forma de menús y/o botones. En el nivel del negocio, el aplicativo es compilado y almacenado en librerías que se puede instalar y/o ejecutar como aplicativo independiente "stand alone". A nivel de datos, se diferencian dos temas importantes. Los datos de entrada y de salida, y la base de datos de registro.

Para los datos de entrada y de salida, en pro del rendimiento y la eficiencia en el procesamiento, se trabajó con datos que están disponibles localmente, y así mismo los resultados del procesamiento son escritos en los discos duros del equipo local. Aunque se entiende que será necesario el uso de recursos y datos de red, se desea evitar la dependencia estos recursos dificultan o y retrasan el geoprocésamiento.

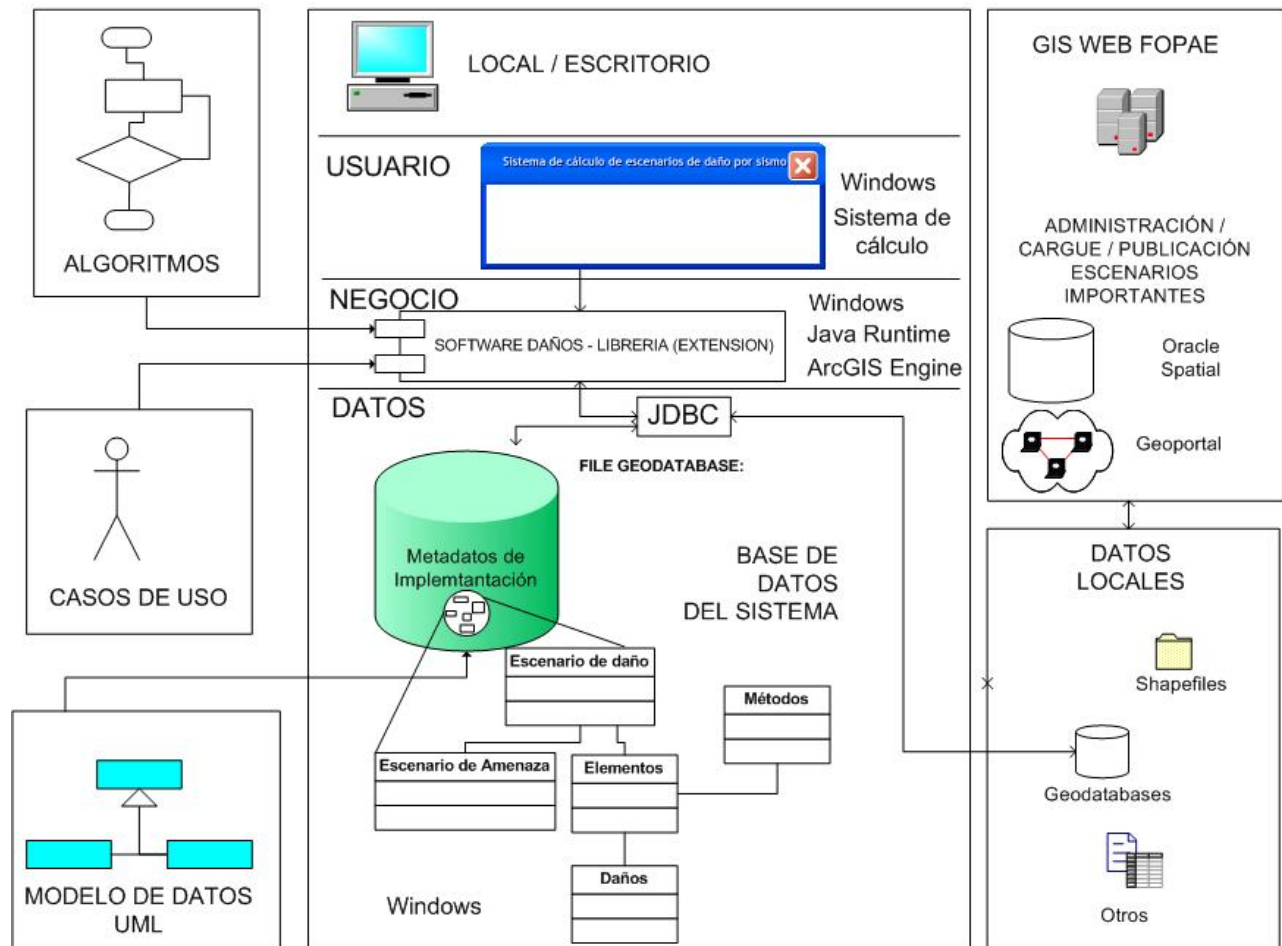


Figura 3.2. Arquitectura general y relación con SIG del FOPAE

Los escenarios de daño generados que se consideran de importancia para la oficina o el sistema de prevención y atención de emergencias se podrán publicar y entregar a partir de los mecanismos y procedimientos ya establecidos en el FOPAE, relacionados con el funcionamiento de su Sistema de Información Geográfico (Web y Stand Alone)

Se ha trabajado el nivel de negocio con ArcGis Engine 9.3, que proporciona librerías de acceso a objetos utilizados por ArcGis Desktop para geoprocetamiento y apertura de archivos y bases de Datos. La **Figura 3.3** muestra como interactúa Java con éstos objetos, específicamente en la manipulación de bases de datos y archivos.

Java Virtual Machine (Máquina virtual de java): Es un programa nativo, capaz de ejecutar instrucciones en código binario especial (el bytecode de java), el cual es generado por el compilador de Java. Es una de las piezas fundamentales de la plataforma Java. Se sitúa en el nivel superior al hardware del sistema sobre el que se ejecuta, y actúa como un puente que entiende tanto el bytecode a ejecutar, como al sistema en que se ejecuta. De esta manera, al escribir una aplicación Java se hace pensando en que se ejecutará sobre una Máquina virtual, y no en un sistema operativo específico, finalmente será la máquina virtual la que se encargue de ejecutar el código Java en el código nativo del dispositivo final.

Aplicación Java: Este componente es la aplicación escrita como tal. Como se observa en la ilustración, ésta se ejecuta sobre la máquina virtual de java, interactuando con Arcgis Engine Runtime por medio de Java Com-bridge. En este componente se encuentran las interfaces, los cálculos matemáticos, el envío y recepción de peticiones a los objetos de ArcGis y la gestión de errores de procesamiento.

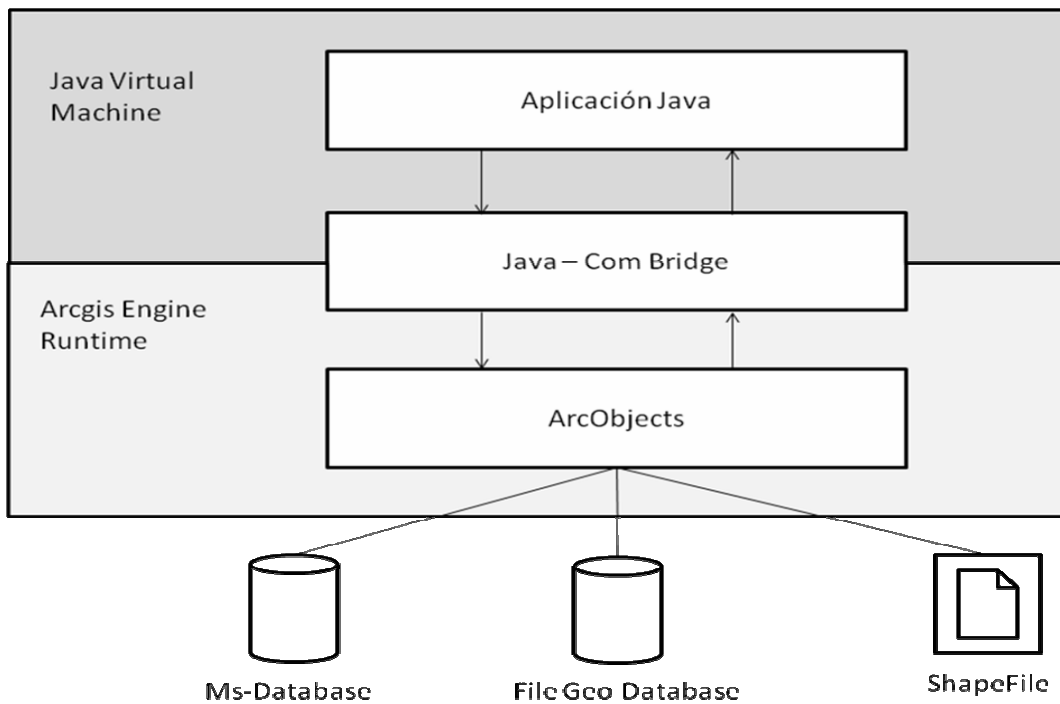


Figura 3.3. Arquitectura de la aplicación

Java COM-Bridge: Es un puente que permite que Java establezca comunicación con los ArcObjects.

ArcObjects: Incluye a todos los objetos que manipulan información geográfica. Este componente se encarga de la lectura y escritura de archivos y bases de datos geográficas, así como del geoprocésamiento, se ejecuta sobre el ArcGis Engine Runtime.

4 METODOLOGÍA DE DESARROLLO APLICADA – RUP

4.1 GENERALIDADES

En el desarrollo de este proyecto se utilizó el Proceso Unificado Racional (RUP por sus siglas en inglés), adaptándolo a las necesidades y características del proyecto. El RUP es una combinación de varias metodologías de diversos autores quienes crearon Rational Software Corporation a mediados de los años 90.

RUP define cuatro fases del ciclo de vida: Inception (concepción), Elaboration (elaboración), Construction (construcción) y Transition (Transición), y una serie de actividades que están presentes en cada una de ellas. El esfuerzo necesario para la ejecución de estas actividades varía de acuerdo a la fase del proyecto en ejecución, como se puede ver en la **Figura 4.1**.



Figura 4.1. Esfuerzo en actividades según la fase del proyecto (Fuente: IBM)

4.1.1 Concepción

El objetivo de esta fase es determinar el alcance del sistema, identificando los requerimientos, y la viabilidad de los mismos, generando además un modelo de casos de uso básico. Al ser una metodología iterativa, la fase de concepción estuvo presente en todo el desarrollo del software. El levantamiento de requisitos se realiza en reuniones periódicas del equipo de trabajo, en reuniones personales con los responsables de suministrar la información de cada requerimiento, o mediante envío de información por correo electrónico.

4.1.2 Elaboración

En la fase de elaboración del sistema se inicia el análisis al problema y definición de la arquitectura del software. UML es una herramienta útil en esta fase; la vista de casos de uso captura el comportamiento de un sistema, subsistema clase o componente tal y como se muestra a un usuario externo. Divide la funcionalidad del sistema en transacciones que tienen significado para los actores (Booch, 2007).

La vista estática modela conceptos del dominio de la aplicación, así como los conceptos internos inventados como parte de la implementación de una aplicación. Esta vista es estática porque no describe el comportamiento dependiente del tiempo del sistema. Los principales componentes de la vista estática son las clases y sus relaciones. (Booch, 2007).

El diagrama de clases y el diagrama de casos de uso producto de esta fase están desarrollados de acuerdo con la especificación 2.0 del lenguaje UML adoptado por el OMG (Object Management Group).

Los patrones de diseño en ingeniería de software también desempeñan un papel importante en esta fase, definiendo aspectos importantes de la arquitectura. En el desarrollo del software se aplicaron los siguientes patrones:

- **Abstract Factory:** El patrón Abstract Factory se utiliza para crear diferentes familias de objetos, en el sistema se utiliza para la creación de lectores de datos de distintos formatos
- **Singleton:** El patrón de única instancia o singleton está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto, su intención es garantizar que la clase solo tenga una instancia y proporcionar un punto de acceso global a dicha instancia. Se utiliza en la creación de objetos que leen información de la base de datos de registro. De esta manera, se garantiza que sólo exista un workspace, y de éste se puedan leer múltiples tablas simultáneamente, sin crear múltiples instancias del workspace en memoria. Igualmente se utiliza al crear hilos para la ejecución de cálculos, evitando que se lancen múltiples procesos de cálculo simultáneamente. Finalmente se utiliza para almacenar en memoria la información de configuración de la aplicación, la cual es cargada al ejecutar la aplicación y se vuelve a escribir en el disco cuando el usuario efectúa un cambio en ella.

4.1.3 Construcción

El objetivo principal de esta etapa es construir el sistema de software, para lo cual esta fase se enfoca en el desarrollo de componentes, interfaces, archivos de entrada y salida del sistema construido. Aquí es donde se efectúa la codificación, partiendo del análisis y diseño de las fases anteriores.

El código escrito además es sometido a pruebas unitarias para verificar que efectúa su tarea correctamente y que interactúa adecuadamente con otras piezas de código. Para estas pruebas se utilizó la herramienta JUnit, generándose una unidad de prueba para cada pieza de código sensible. El producto resultante de esta fase es el código fuente del prototipo.

- **Convenciones de código fuente**

Para la codificación de algoritmos y desarrollo de interfaces se buscó seguir las recomendaciones que se mencionan en Code Conventions for the Java Programming Language (documento disponible en <http://java.sun.com/docs/codeconv/>), procurando que los nombres de variables, clases, interfaces y métodos estén en inglés, ya que las clases, interfaces y métodos propios Java, y aquellos propios del Arcgis Engine se encuentran en ese idioma. Gracias a la adopción de estas convenciones se garantiza la legibilidad del código y la fácil interpretación del mismo por parte de cualquier programador que lo revise.

- **Documentación del código fuente**

Además de las convenciones en la escritura de código fuente, otra herramienta útil para la comprensión de código de terceros es la documentación técnica. JavaDoc es una herramienta para documentar código fuente en formato HTML directamente desde el código.

Al documentar el código fuente siguiendo el estándar JavaDoc, se genera un documento HTML que describe los paquetes, clases, interfaces, métodos y variables que componen la aplicación. La información del documento se visualiza desde un navegador web, y aparece organizada por paquetes, con un índice de clases, interfaces y excepciones.

El documento resultante se visualiza óptimamente desde un navegador web, y no está diseñado para leerse de principio a fin, sino siguiendo los enlaces que el lector desee seguir. El documento impreso luce caótico y desordenado, perdiendo su navegabilidad. La documentación puede observarse en la carpeta Documentación\JavaDoc\ que hace parte del prototipo

El resultado de esta fase es el código fuente del prototipo, las clases de pruebas unitarias generadas, y la documentación técnica en formato JavaDoc. El código ejecutable está escrito en lenguaje Java, y está configurado como proyecto para el entorno de desarrollo Eclipse, aunque puede llevarse sin problema a otros entornos como NetBeans y JDeveloper.

El código puede compilarse con versiones de Java 5 o posterior, con versiones anteriores no compilará ni ejecutará.

4.1.4 Transición

El objetivo de la fase de transición es llevar el producto de software de desarrollo a producción, haciéndolo disponible al usuario final. Las actividades de esta fase incluyen el entrenamiento a los usuarios y las pruebas beta; al ser el producto desarrollado en la fase I un prototipo, la fase de transición no ha sido completada.

4.2 DOCUMENTACIÓN DEL SISTEMA CON LOS FORMATOS ESTABLECIDOS POR LA ENTIDAD

La Entidad dentro de sus procesos de mejoramiento y calidad ha definido el proceso para creación de software, aunque este está más dirigido a las aplicaciones web que constantemente desarrolla el grupo de informática, se guarda el mismo enfoque que una aplicación de escritorio, a continuación se resumen las características generales de estos pasos para la programación y en el **Anexo 1 a 4** se encuentran los formatos debidamente diligenciados.

4.2.1 Levantamiento de requisitos funcionales

El grupo de escenarios de daño define en reuniones qué tareas debe cumplir la aplicación y se asigna a diferentes integrantes la redacción de un requerimiento formal. Finalmente, el requerimiento es revisado nuevamente en reunión y es enviado para el desarrollo. Los requerimientos pueden ser modificados a medida que el software sea desarrollado y se tenga una visión más específica de cada módulo a desarrollar. Como resultado de esta labor se obtuvieron los formatos ADM-FT-16 que se encuentran en el **Anexo 1**, el formato y ADM-FT-17 no aplica dado que no se ha hecho seguimiento de los requisitos a la fecha.

4.2.2 Análisis de requisitos funcionales

Una vez el requerimiento es levantado, éste pasa a manos del desarrollador de software quien efectúa el análisis respectivo desde el punto de vista informático. Los detalles referentes a fórmulas matemáticas y origen de información son discutidos con el grupo. El resultado de esta fase se refleja en los formatos ADM-FT-18 del **Anexo 1**.

4.2.3 Casos de uso

Mediante el análisis de requisitos funcionales se llega a identificar los casos de uso del prototipo. Aunque también se logra percibir que la aplicación final tendrá muchos más casos de uso, solo se tienen en cuenta aquellos que son cobijados por el alcance del prototipo a desarrollar en la fase I. Los casos de uso identificados se ilustran en el formato ADM-FT-19 Y ADM-FT-42 del **Anexo 2**.

4.2.4 Análisis de casos de uso

El análisis de los casos de uso identificados se encuentra en los formatos ADM-FT-20 del **Anexo 2**.

4.2.5 Modelo de datos

Los módulos del modelo de datos creados para ejecutar análisis de riesgo sísmico, como edificaciones, red de acueducto y alcantarillado principal y secundaria, tanques, puentes, cables colgantes y enterrados, y red de gas, utilizan y dependen de otros módulos como amenaza, investigaciones de vulnerabilidad y parametrización de métodos que son transversales y permiten que los análisis de riesgo se configuren para ejecutarse en un marco de referencia común que se integra en el concepto de escenario de daño (ver **Figura 4.2**). A continuación se explica el funcionamiento de un método cualquiera de análisis ejecutado sobre cualquier categoría. Los diagramas del modelo de datos de cada una de las secciones se presentan en el formato ADM-FT-22 del **Anexo 3**.

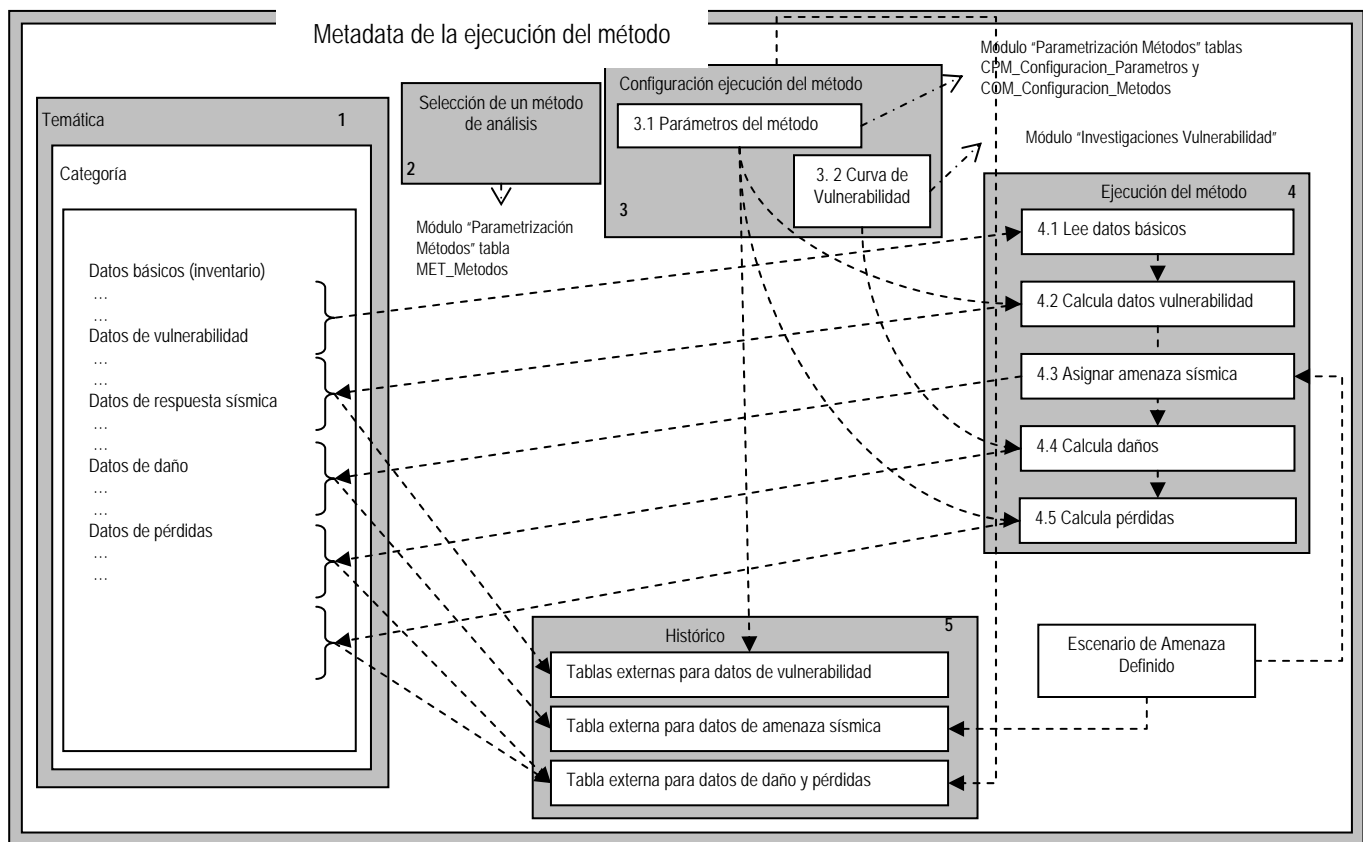


Figura 4.2. Esquema de trabajo del modelo de datos espaciales

El diagrama anterior tiene el siguiente funcionamiento:

- 1 Selección de una categoría que se le aplica un método particular. Esta selección se hace bajo la estructura jerárquica temática, categoría.
- 2 Selección de un método de análisis aplicable a la categoría seleccionada en el paso anterior. Para esto se utiliza la sección de la base de datos llamada "Parametrización Métodos"; específicamente la tabla "MET_Metodos"
- 3 Configuración de la ejecución del método seleccionado, que se subdivide en:

- 3.1 Configuración de los parámetros del método: Usando la sección de la base de datos llamada "Parametrización Métodos" la cual funciona para configurar un método de forma global con todos sus parámetros (tabla COM_Configuracion_Metodos).
- 3.2 Selección de la investigación de vulnerabilidad utilizando la sección "Investigaciones Vulnerabilidad" de la base de datos, la cual relaciona simultáneamente las curvas de vulnerabilidad que aplican en cada categoría y en el caso de familias de curvas de vulnerabilidad que aplican a tipología dependiendo del método de análisis.
- 4 Ejecución del método, en esta se identifican los siguientes procesos.
 - 4.1 Seleccionar los datos organizados y preparados para aplicar el método.
 - 4.2 Utilizando los datos de 4.1 y la configuración realizada en 3.1 (parámetros del método) se calculan y se almacena en la base de datos de la categoría los datos relacionados con la vulnerabilidad de la categoría.
 - 4.3 Utilizando un escenario de amenaza particular se asigna a la categoría en evaluación los datos de la respuesta sísmica.
 - 4.4 Utilizando el producto de 4.1 si aplica en para el método, la asignación de amenaza de 4.3 y la selección realizada en 3.2 (curva de vulnerabilidad) se calculan a la categoría en evaluación los datos relacionados con los daños.
 - 4.5 Utilizando 4.1 (datos básicos), 4.4 (datos de daño), y la configuración de los parámetros del método realizada en 3.1 (opcional), se calculan los datos de pérdidas y afectaciones.
- 5 Almacenamiento de los datos históricos para la ejecución del método con diferentes configuraciones. Se identifican tres tipos de almacenamiento.
 - 5.1 Almacenar los datos de vulnerabilidad en una o varias tablas externas. Los datos almacenados en 4.2 se almacenan en una o varias tablas teniendo en cuenta la configuración realizada en 3.1 (parámetros de método). Si se desea almacenar datos finales de vulnerabilidad se crea una tabla externa para almacenar el histórico y esta tabla dependerá de la configuración de los métodos (COM_Configuracion_Metodos).
 - 5.2 Almacenar los datos de respuesta sísmica en una tabla externa. Los datos resultantes de 4.3 se almacenan en otra tabla teniendo en cuenta el código del escenario de amenaza que se esté analizando.
 - 5.3 Almacenar los datos de daño y pérdidas en una tabla externa. Los datos resultantes de 4.4 y 4.5 se almacenan en otra tabla teniendo en cuenta la configuración del método que se esté ejecutando, es decir los identificadores que se utilizan en 3: configuración de los parámetros del método (COM_Configuracion_Metodos) y configuración de la investigación de vulnerabilidad (INV_Investigaciones_Vulnerabilidad).

El diagrama descrito con anterioridad (que funciona algunas veces de manera simplificada - algunos procesos no son requeridos o se simplifican-, por ejemplo en la aplicación de un método de análisis de daños que no requiera datos intermedios de vulnerabilidad, o no calcule datos de pérdidas) se integra cuando se trabaja de manera repetitiva para varias categorías. El módulo de escenario de daño no es más que todo el proceso descrito en el diagrama anterior, aplicable a diferentes categorías, e integrado en un nombre particular de escenario de daño (ver **Figura 4.3**).

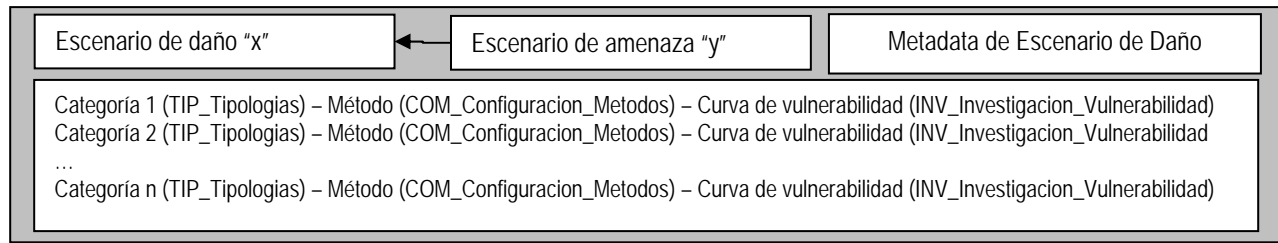


Figura 4.3. Proceso de configuración de métodos

Esta integración permitirá en fases posteriores implementar el requisito funcional de evaluación del riesgo teniendo en cuenta los escenarios de daño de los diferentes elementos expuestos y los indicadores de capacidad de respuesta institucional y población. También permitirá implementar interfaces genéricas más amigables que reflejen mejor el concepto de escenario de daños y escenarios de riesgo considerando la ciudad como un sistema urbano integrado. Esta interfaz que puede funcionar a manera de vista de árbol o “treeview” debería generarse de manera automática y construirse con base en la organización del módulo “escenario de riesgo”. Como se ilustra en la **Figura 4.4**.

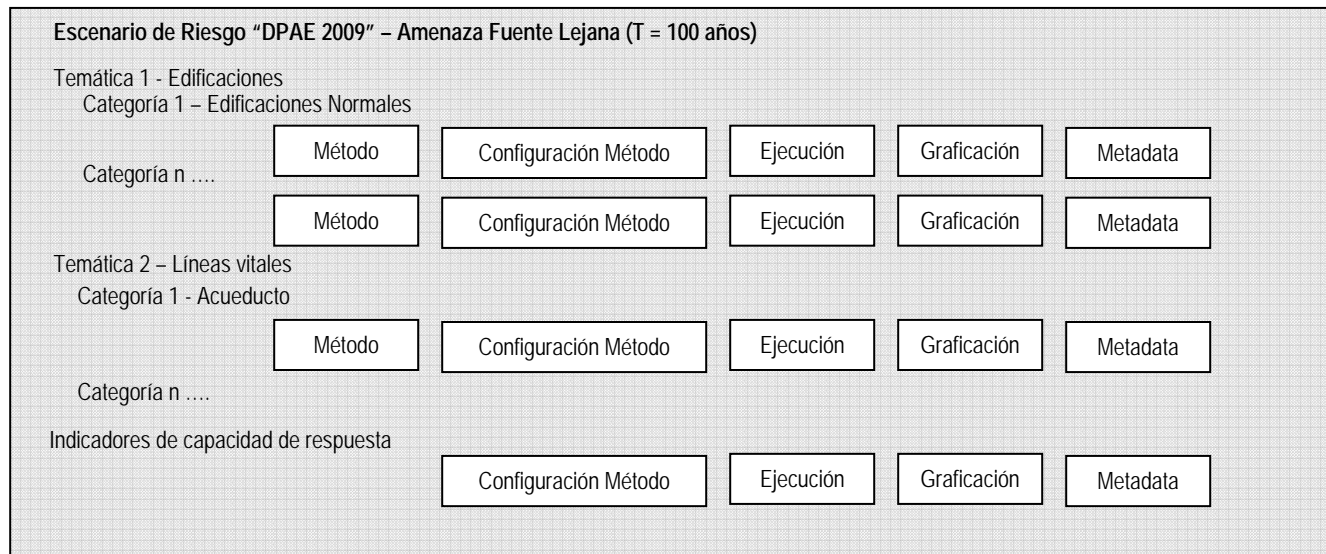


Figura 4.4. Árbol de procesos de un escenario de riesgo

El trabajo realizado durante esta fase del proyecto fue principalmente la creación de las tablas vacías existentes en el modelo de datos, para facilitar el proceso de programación, el cargue de datos en las tablas de las secciones de curvas de vulnerabilidad y amenaza, la definición de los dominios de algunos de los campos de la base de datos y la normalización de las coberturas de información geográfica a los estándares del modelo de datos, el resultado obtenido es el modelo de datos espacial físico es la filegeodata base nombrada sistema de calculo (ver **Figura 4.5**).

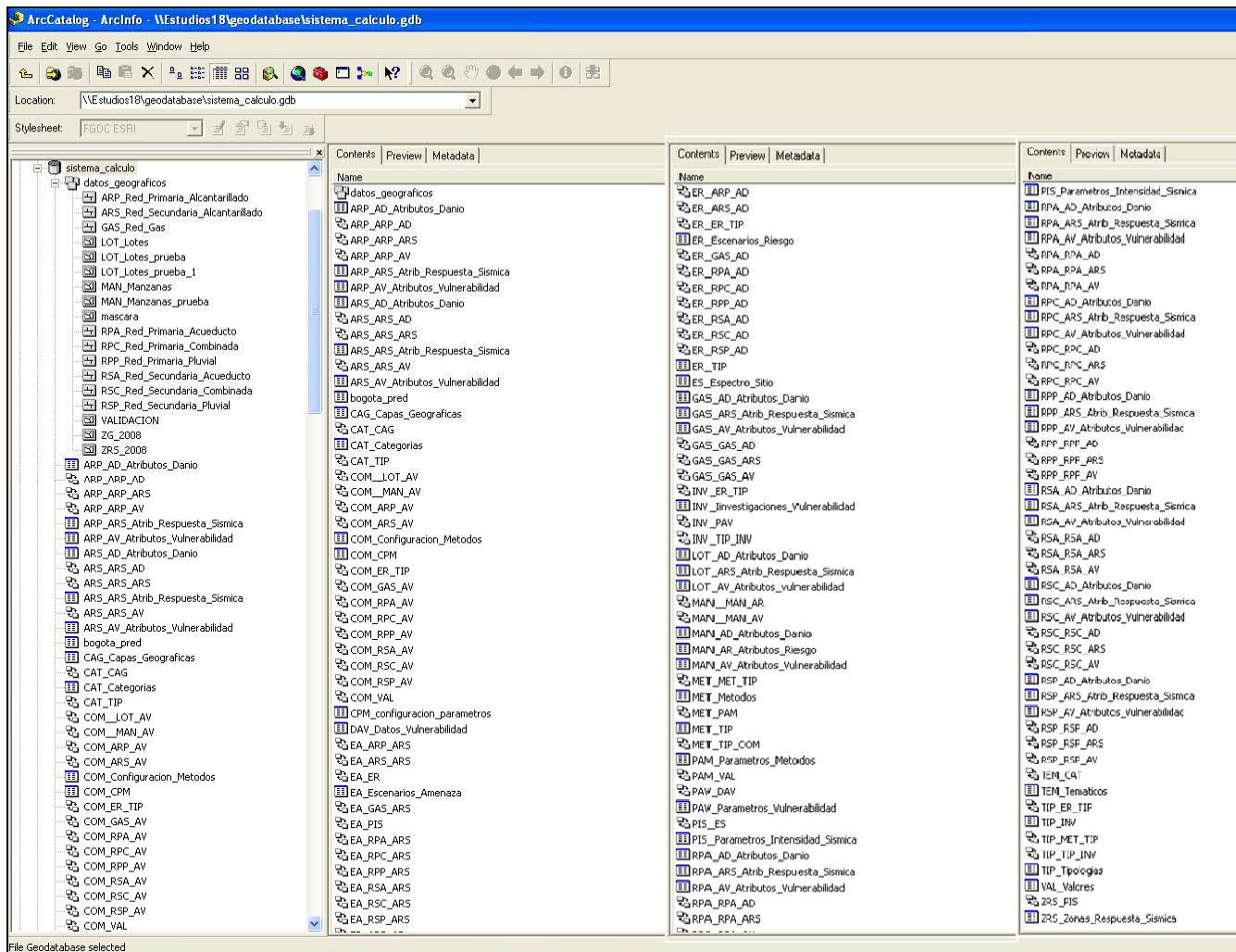


Figura 4.5. Base de datos geográfica del sistema de cálculo según el modelo de datos

4.2.6 Diccionario de datos

El archivo base del diseño generado sobre Visio 2003 se documentó detalladamente utilizando el valor etiquetado “documentation” para campos y clases (tablas y features classes). Esta documentación no se traslada de manera automática a la Geodatabase por cuanto no existen directamente en los objetos de la geodatabase propiedades para las descripciones o la documentación.

A partir de la Geodatabase o el XML Workspace se generaron diferentes versiones de documentación y diccionario de datos de la geodatabase. Los archivos finales se presentan en Html y Excel que se anexan al formato ADM-FT-23 en el **Anexo 3**.

4.2.7 Despliegue de la aplicación

Inicialmente se tenían dos opciones para desplegar la aplicación:

- Como un complemento para ArcGis que se ejecutaría dentro de ésta aplicación
- Como una aplicación independiente que utilizaría objetos de ArcGis Engine.

Para proteger la información de manipulación accidental desde el entorno ArcGis, y debido a la selección de lenguaje Java como lenguaje de programación del producto, se eligió la segunda opción. (Arcgis solo permite integrar en su interfaz módulos desarrollados con la plataforma .net de Microsoft). La aplicación se desplegará como un archivo bytecode que se ejecutará por una máquina virtual de Java.

Todas las clases y archivos de configuración desarrollados en la fase de programación se encuentran en un único archivo JAR; las librerías y objetos externos de Arcgis se almacenan en archivos JAR independientes, que deberán estar presentes para permitir la ejecución. No hay restricciones acerca de qué carpeta utilizar para almacenar el archivo JAR ejecutable.

El directorio de archivos donde se instala la aplicación luce de esta manera:

[Unidad:]\[Directorio de instalación].\SistemaCalculo.jar

[Unidad:]\[Directorio de instalación].\LIB\[Archivos Jar Externos]

El prototipo se conecta a una base de datos GDB que preferiblemente debe encontrarse almacenada en el mismo equipo local donde se ejecuta la aplicación, aunque puede trabajar aún si la base de datos sólo se encuentra disponible a través de la red.

El prototipo no requiere de servidores de aplicaciones para ejecutarse, por lo tanto el formato dispuesto por la entidad para despliegue de aplicaciones web no aplica.

5 DESCRIPCIÓN DEL PROTOTIPO DEL SISTEMA

En esta sección se describe el funcionamiento de la aplicación y los módulos básicos que la componen. Es importante aclarar que actualmente la aplicación se encuentra en estado de prototipo y no está finalizada, que algunos de los módulos que aquí se describen solamente se encuentran en fase de diseño y no han sido desarrollados. Solamente se describen con mayor detalle aquellos módulos que están más avanzados, o que están lo suficientemente avanzados como para generar resultados visibles. Sin embargo, todos estos módulos e interfaces serán revisados y mejorados en la siguiente fase del proyecto.

5.1 REQUERIMIENTOS DE SOFTWARE

Sistema operativo:

Microsoft Windows Server 2003 o 2008

Microsoft Windows Vista, XP, o 2000

Red Hat (Enterprise Linux) AS/ES

SUSE Linux Enterprise Server

Sun Solaris

Arcgis Engine Runtime: Versión 9.3.

Java Virtual Machine: Versión 1.5 o posterior.

Requerimientos de Hardware:

Procesador Pentium IV de 2.6 Ghz o Superior.

Memoria Ram de 2 Gb.

15 Mb de espacio en Disco (Para almacenar la aplicación)

15 GB de espacio en Disco (Para almacenar los datos utilizados por la aplicación)

5.2 LANZAMIENTO DE LA APLICACIÓN

La aplicación ha sido desarrollada en lenguaje Java, utilizando los objetos ArcObjects que Arcgis Engine 9.0 proporciona. Java es un lenguaje multiplataforma, y las aplicaciones desarrolladas en él se ejecutan en sistemas operativos Linux, Windows y Mac y demás sistemas que cuenten con una máquina virtual de Java (JVM por sus siglas en inglés), sin embargo Arcgis Engine Runtime no está disponible para MacOS. lo que hace que la aplicación no pueda ejecutarse en ese sistema operativo.

Para ejecutar la aplicación se debe dar doble clic en el archivo "SistemaCalculo.jar". Esto debería lanzar la aplicación automáticamente.

Si la máquina virtual no está instalada, o está mal configurada, aparecerá un mensaje de Windows solicitando asociar un programa para abrir archivos de extensión jar, como se ilustra en la **Figura 5.1**. Pantalla "Abrir con"..

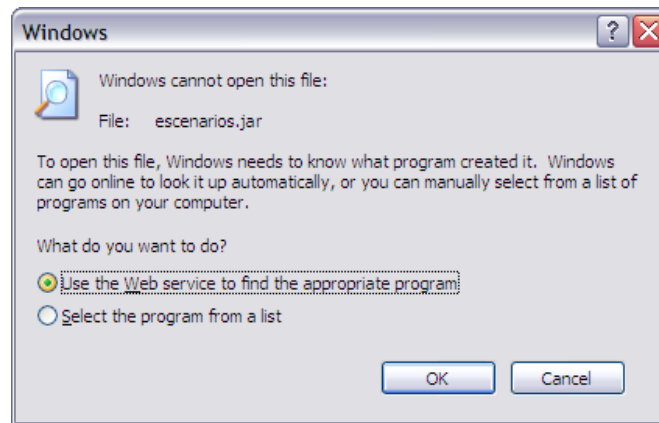


Figura 5.1. Pantalla "Abrir con".

Si esto sucede debe verificar que la máquina virtual está instalada, y que corresponde a la versión 1.5 o superior. Para comprobar la versión de su máquina virtual siga este procedimiento: (en Windows XP)

Ejecute la consola de comandos de DOS. (Puede hacerlo dando clic en inicio, ejecutar y escribiendo "cmd" en la ventana que aparece a continuación).

Escriba el siguiente comando en la línea de comandos:

```
java -version
```

Si la máquina virtual está instalada aparece un mensaje similar al de la **Figura 5.2**. Verificación de la versión de Java en la consola de DOS

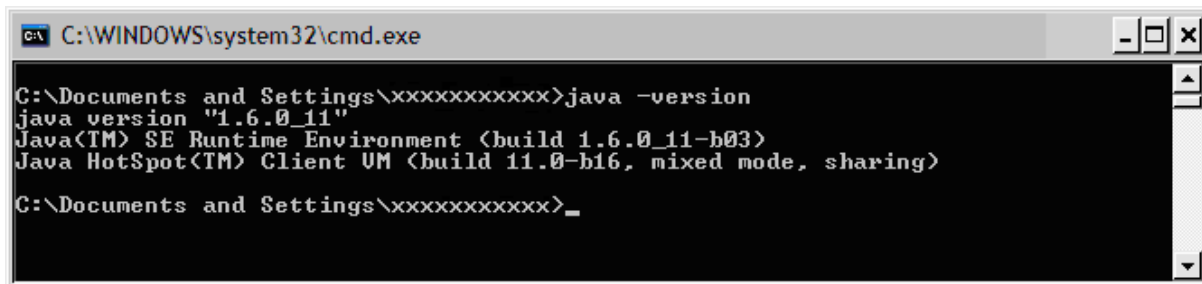


Figura 5.2. Verificación de la versión de Java en la consola de DOS

Si la máquina virtual no está instalada, aparece un mensaje como el siguiente:

'java' no es reconocido un comando interno o externo, programa o archivo batch

En caso de que la máquina virtual no esté instalada o la versión sea inferior a la 1.5 debe descargarla e instalarla. (Puede descargarla desde la dirección <http://java.com/en/download/manual.jsp>)

Si la versión de java es correcta y aún así no logra lanzar la aplicación dando doble clic en el archivo jar, intente escribiendo el siguiente comando en la línea de comandos:

```
java -jar SistemaCalculo.jar
```

Tenga en cuenta que debe escribir este comando desde la carpeta que contiene el archivo jar.

5.3 PANTALLA PRINCIPAL

La aplicación es una aplicación MDI (Multiple Document Interface); desde la pantalla principal se tiene acceso a todas las opciones de la aplicación, y en ella se pueden contener otras ventanas y documentos.

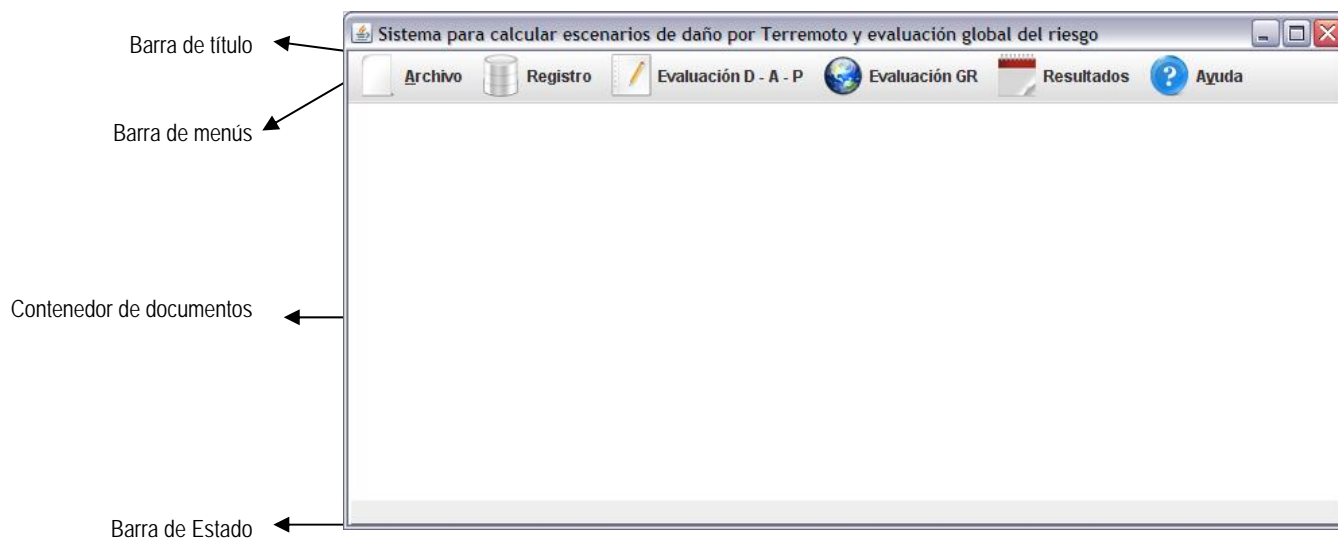


Figura 5.3. Pantalla Principal

Barra de Título: Muestra el nombre de la aplicación. Muestra los botones estándar de Windows: Minimizar, Restaurar y Cerrar.

Barra de menús: Muestra el menú de opciones, desde donde se acceden a los módulos de la aplicación.

Contenedor de Documentos: Espacio que inicialmente aparece vacío, donde se puede visualizar el mapa de resultados y la consola de ejecución.

Barra de estado: Muestra información sobre el estado actual de la aplicación.

5.4 SECCIONES

Módulo de Registro: Permite registrar la información de entrada que utiliza la aplicación; está dividida en categorías, como se observa en la Figura 5.4. Menú de Opciones del Módulo de Registro. Actualmente el módulo de registro no se encuentra totalmente desarrollado: aunque los módulos de cálculo del prototipo hacen uso de la base de datos de registro para cargar la información registrada, actualmente no es posible registrar nuevos orígenes de datos.



Figura 5.4. Menú de Opciones del Módulo de Registro

El módulo de registro está diseñado de tal manera que sea pueda agregar archivos de diferentes formatos; el usuario puede elegir entre bases de datos Access, Shape Files, o Personal GeoDatabases, y una vez seleccionado el archivo, debe registrar las tablas y los campos que contienen información útil a los algoritmos de cálculo.

Módulo Daños, Afectaciones y Pérdidas: En éste se realizan los cálculos partiendo de la información registrada. Está dividido en varios módulos de los cuales se encuentran desarrollados:

Edificaciones / Índice de vulnerabilidad para edificaciones: Calcula el índice de vulnerabilidad para todas las edificaciones registradas en la tabla unificada.

Edificaciones / Daños y pérdidas económicas para edificaciones: Calcula el porcentaje de daños en edificaciones, su pérdida económica, y la afectación en la población, partiendo del índice de vulnerabilidad calculado y de la información registrada para características locales.

Líneas Vitales / Tasa de roturas en líneas vitales: Calcula la tasa de roturas por kilómetro para líneas vitales. Requiere registro previo de líneas de servicios y/o movilidad, y amenazas.

Población / Víctimas, heridos y afectados: El menú estaba pensado para ejecutar el cálculo de afectaciones en la población. Sin embargo, este proceso se está desarrollando paralelamente al cálculo de daños y pérdidas económicas (ver numeral 6.2)

Los demás módulos se encuentran en proceso de especificación, o diseño.



Figura 5.5. Menú de Opciones del Módulo de Daños, Afectaciones y Pérdidas

El usuario puede ejecutar solamente un proceso de cálculo a la vez. Cualquier intento de iniciar la ejecución de un segundo proceso de cálculo mientras ya esté uno en ejecución es ignorado.

Al comenzar cualquiera de los procesos de cálculo se mostrará la consola de ejecución (descrita más adelante), que ofrece información útil sobre el proceso en ejecución.

Consola de ejecución: Dependiendo del área cálculo y la cantidad de información, los procesos de cálculo pueden tomar mucho tiempo. Para que el usuario pueda ver información sobre el progreso de los cálculos y errores que se puedan presentar en los mismos se creó una consola de ejecución. La información desplegada en ella varía de acuerdo al tipo de cálculo que esté en ejecución. La consola se abre automáticamente cuando inicia un proceso de ejecución de cálculos, y está localizada en la parte inferior de la pantalla.

La información que despliega la consola depende del cálculo que se realice. Se puede configurar la consola para que despliegue información sobre los errores ocurridos en la ejecución del programa. La información del error incluye una breve descripción del error ocurrido, y también puede incluir información útil para el programador en tareas de depuración.

Log de monitoreo: La velocidad de ejecución hace imposible la mayoría de las veces apreciar con detenimiento los mensajes de error que puedan producirse en la consola de ejecución, por lo tanto todos los mensajes de error son almacenados en un archivo plano que puede visualizarse desde cualquier editor de texto, como el bloc de notas o WordPad. En este archivo se almacenan sólo los mensajes de error que se producen en la ejecución de cálculos, y adicionalmente puede incluirse la información de trazado de pila del error. En la mayoría de los casos mensaje proporciona información suficiente para que el usuario identifique la naturaleza del error; que normalmente se origina por problemas con los datos de entrada.

Un ejemplo de mensaje de error es el siguiente:


```
Registro omitido.Excepción arrojada :  
El objeto no tiene la información mínima para ejecutarse:  
estrato = 0  
uso = OTROS  
Año de construccion = 0  
Resgistro de diseño = false  
Tipologia estructural = null  
Divisiones = null  
Entrepiso = null  
Numero de pisos = null  
Pendiente del terreno = null  
Área construida = 0.0  
Área del terreno = 8769.88  
Empuje de Tierras = null  
Irregularidades en planta = null  
Irregularidades en altura = null
```

Esta información es fácilmente entendible por el usuario de la aplicación, quien rápidamente puede determinar la causa del error y tomar las medidas pertinentes para solucionarlo o pasarlo por alto. Sin embargo, algunos errores pueden originarse por causas desconocidas o por falencias que no tienen que ver con los datos de entrada, como falta de memoria, o la imposibilidad de acceder a un recurso. Para hacer seguimiento a estos casos excepcionales se permite incluir en el archivo log la información del trazado de pila, que muestra al programador un mensaje con la descripción del error, una lista de los métodos invocados, indicando además la línea de código y el archivo fuente donde se produce la excepción. Un ejemplo de trazado de pila es el siguiente:

```
org.dpae.scenarios.calculation.vulnerabilityIndex.VulnerabilityIndex.calcular  
IV(VulnerabilityIndex.java:131)  
org.dpae.scenarios.calculation.CalcLauncher.initVICalc(CalcLauncher.java:7)  
org.dpae.scenarios.calculation.CalcLauncher.run(CalcLauncher.java:175)  
java.lang.Thread.run(Unknown Source)
```

Esta información resulta de utilidad para el programador a la hora de depurar la aplicación.

Módulo Evaluación Global del Riesgo: El módulo se encuentra en etapa de diseño.



Figura 5.6. Menú de Opciones de Evaluación Global del Riesgo

Módulo Resultados: Permite visualizar los resultados de los cálculos en mapas. Requiere la ejecución previa de los cálculos a visualizar. El módulo se encuentra en etapa de diseño.



Figura 5.7. Menú de opciones del módulo de Resultados

Módulo Ayuda: Muestra el manual en línea del usuario, o información sobre el desarrollo del proyecto. El manual del usuario no está terminado (depende del desarrollo de los módulos faltantes), pero será un documento HTML que se visualizará en un navegador externo a la aplicación.

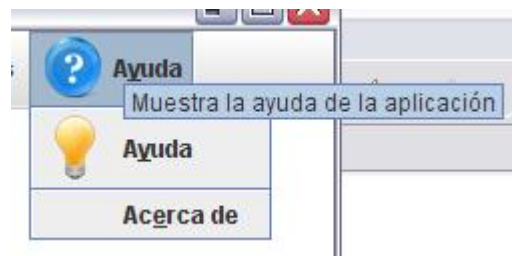


Figura 5.8. Menú Ayuda de la Aplicación

6 REALIZACIÓN DE PRUEBAS Y RECOMENDACIONES

Se ejecutaron pruebas a cada uno de los módulos desarrollados para este prototipo. Las pruebas se realizaron desde la interfaz del prototipo, o mediante pruebas unitarias de JUnit. Gracias a las pruebas se detectaron algunos errores de lógica en los algoritmos, o de codificación.

Todos los errores detectados fueron corregidos; gran parte de los errores eran demasiado específicos y muy fáciles de resolver. Los hallazgos más importantes de las pruebas dejaron ver problemas de diseño más que errores de algoritmia o codificación.

6.1 PROCEDIMIENTOS MEJORADOS DURANTE EL DESARROLLO

El primer problema de gran magnitud que tenía un fuerte impacto en el rendimiento de la aplicación tenía que ver con la lectura de información registrada que se utilizaría en los algoritmos de cálculo. Con el diseño inicial se permitiría registrar la información de múltiples bases de datos.

Al estar almacenadas en diferentes tablas y bases de datos, la información de cada elemento debía ser recopilada en el momento de ejecutar los cálculos. Entre más tablas y bases de datos fueran registradas, más tiempo se tardaría el prototipo en realizar una corrida; por lo tanto si se quisiera ejecutar una corrida completa de cálculo de índice de vulnerabilidad sobre las edificaciones de Bogotá, con la información dispersa en 2 tablas, la velocidad de ejecución decaería notablemente.

Partiendo de esta problemática (y sin tener en cuenta otras desventajas secundarias que ocasiona este esquema), se modificó el sistema de registro de datos para que toda la información de entrada sea almacenada en un único archivo y una única tabla en el momento de ser registrada (denominada tabla unificada en los algoritmos).

Las **Figura 6.1** y **Figura 6.2** ilustran la situación anterior y la actual con la que trabaja el sistema.

Las líneas punteadas en la **Figura 6.1** y la **Figura 6.2** ilustran cuántas tablas deben ser accedidas en el momento de efectuar el cálculo. Se observa claramente que el nuevo esquema de registro accede a una sola tabla; consumiendo menos memoria y agilizando el proceso.

También se encontró un problema de diseño con la base de datos de registro de parametrización de métodos. El modelo inicial de base de datos no reflejaba relaciones entre los métodos y sus parámetros, y además permitía la reutilización de un parámetro a un nivel más detallado del que requería el proyecto, creando una serie de tablas innecesarias que complicaban la programación, reducían la legibilidad del modelo y no aportaban beneficio alguno.

Después de varias reuniones y evaluar varias propuestas, se llegó a un mejor modelo que es el presentado actualmente. Sin embargo, este modelo aún puede ser sometido a modificaciones según las siguientes fases de desarrollo.

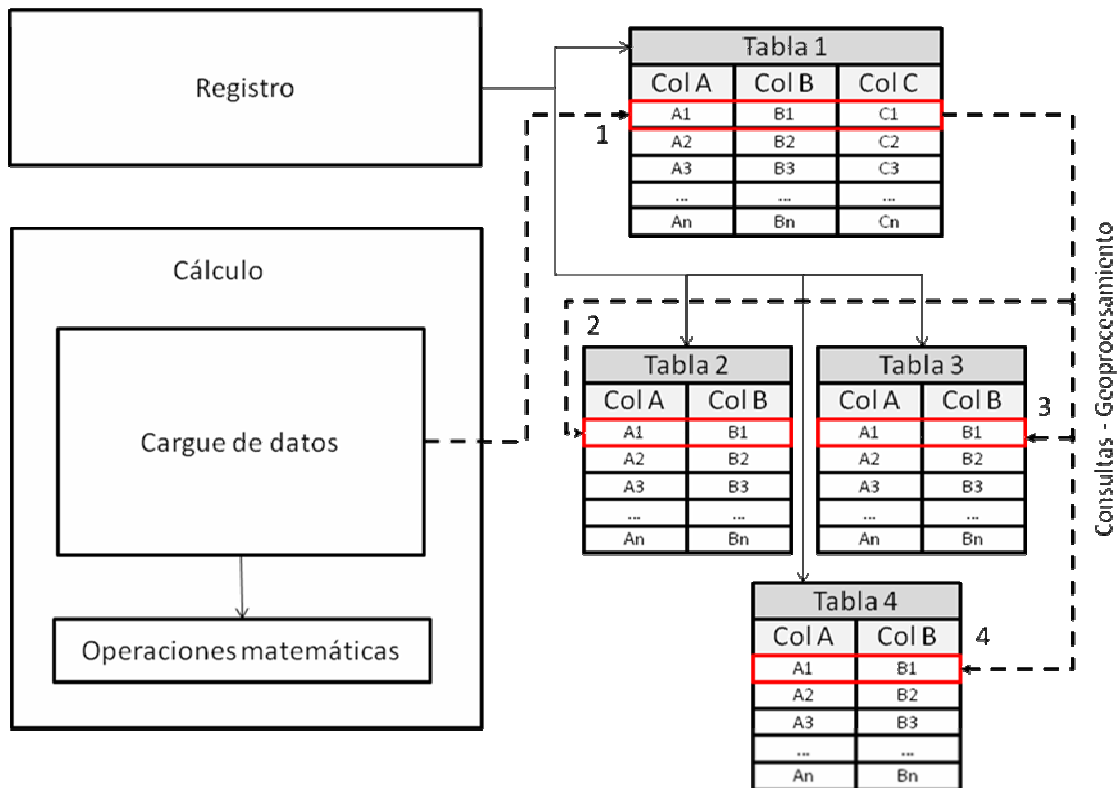


Figura 6.1. Esquema de registro anterior

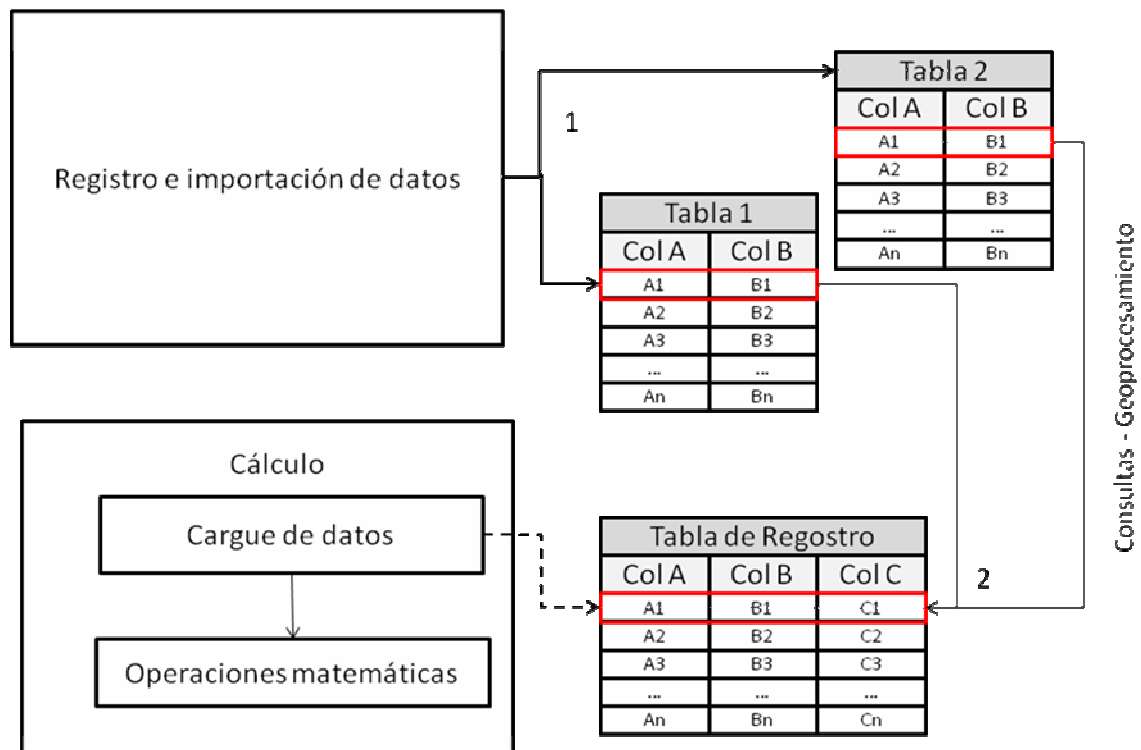


Figura 6.2. Esquema de registro actual

El último error de diseño estaba en el algoritmo de evaluación de daños en edificaciones. El diagrama de flujo presentado por el especialista no mencionaba el orden en que debían leerse los datos de lotes y manzanas. Inicialmente se diseñó el módulo para cargar una manzana, asociarla a todas las edificaciones que contiene y efectuar los cálculos. Sin embargo, debido al gran número de edificaciones y manzanas, la asociación entre estos elementos era “demasiado lenta”. Nunca se terminó de ejecutar el algoritmo completo, pero según los cálculos realizados, tardaría cerca de 9 días en terminar la corrida. Luego de varias iteraciones y mejoras, se invirtió el orden de asociación, y se diseñó un código para almacenar promedios en la tabla de manzanas. Finalmente se redujo el tiempo de ejecución a tan solo 10 horas.

Es más eficiente y organizado llevar toda la información recopilada a un único archivo o *tabla unificada* desde donde se tiene acceso inmediato a todos los atributos de un elemento analizado. Obtener la información desde diferentes archivos o tablas reduce drásticamente la ejecución de cálculos; por ello es necesario reducir al máximo el número de tablas utilizadas para almacenar y relacionar la información.

La información que se utiliza muchas veces durante un proceso de ejecución (por ejemplo, curvas de vulnerabilidad en cálculo de índice de daño) debe ser cargada en memoria la primera vez que se utilice, y mantenerse en ella hasta que termine la ejecución del algoritmo para todos los elementos analizados. Aunque con este esquema se mantiene menos memoria disponible, se obtiene una reducción en el tiempo que toma ejecutar los algoritmos, ya que cargar dicha información una y otra vez toma mucho tiempo.

Finalmente, se ha observado que a pesar de la claridad y detalle con que los especialistas presentan los algoritmos de cálculo éstos pueden ser modificados por el desarrollador. Dichas modificaciones nunca alteran el resultado final de las funciones y fórmulas utilizadas, sino que buscan optimizar el algoritmo para que se ejecute de manera más eficiente. De todos modos, dichas modificaciones son resultado de pruebas y son sometidas a discusión.

A continuación se ilustra de la **Figura 6.3** a **Figura 6.6** los resultados del módulo de edificaciones y líneas vitales obtenidos por el prototipo del sistema, se reitera que son preliminares y sujetos a cambios, pero con ellos se visualizaron algunas recomendaciones y mejoras que se realizarán en la siguiente fase del proyecto.

6.2 RECOMENDACIONES PARA LA SEGUNDA FASE DEL PROYECTO

- En cuanto a la interfaz de la pantalla principal se recomienda crear un menú para evaluación de vulnerabilidad (aquellas características de los elementos expuestos que no dependen de la amenaza). De esta manera el usuario puede percibir más intuitivamente cómo funciona el flujo de ejecución. Al implementar este menú, desaparecerían del menú Evaluación D – A – P las opciones “índice de vulnerabilidad” de los submenús “Edificaciones” y “Edificaciones indispensables”.
- Como la ejecución de una corrida es un proceso tan extenso, y éste no puede ser pausado (si se interrumpe la ejecución toca volver a comenzar desde el principio), es necesario advertir al usuario sobre el tiempo que tomará la aplicación. Además de las pantallas de configuración de una corrida que están previstas para la siguiente fase de desarrollo, deben diseñarse unas pantallas de advertencia que se mostrarán justo antes de iniciar los procesos de cálculo.
- Finalmente es recomendable explorar la posibilidad de programar utilizando los objetos “Relationship” de ArcObjects, para evaluar si éstos ofrecen un mejor rendimiento que el obtenido actualmente mediante consultas SQL a la hora de relacionar información de diferentes tablas.

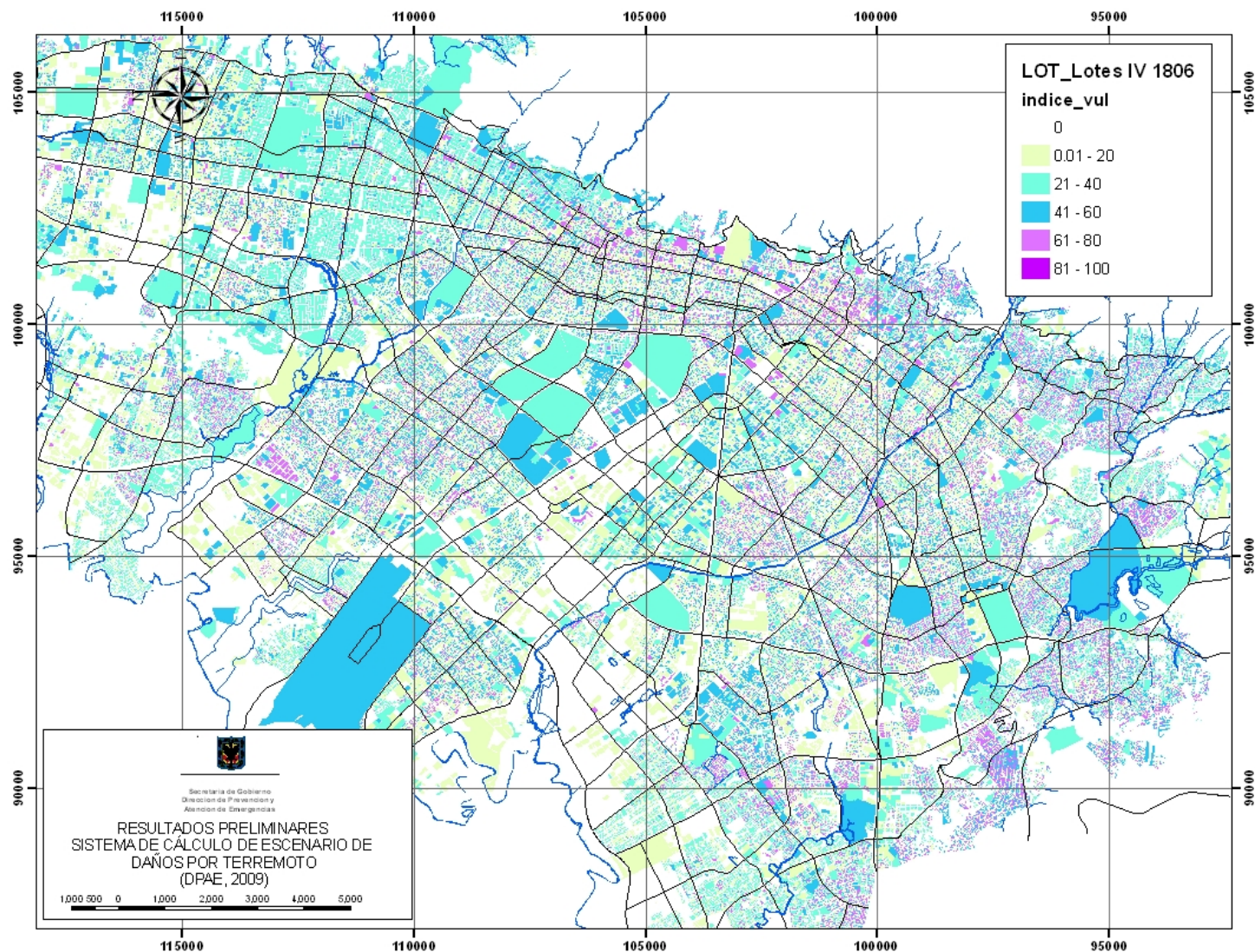


Figura 6.3. Resultados preliminares del prototipo de índice de vulnerabilidad de edificaciones

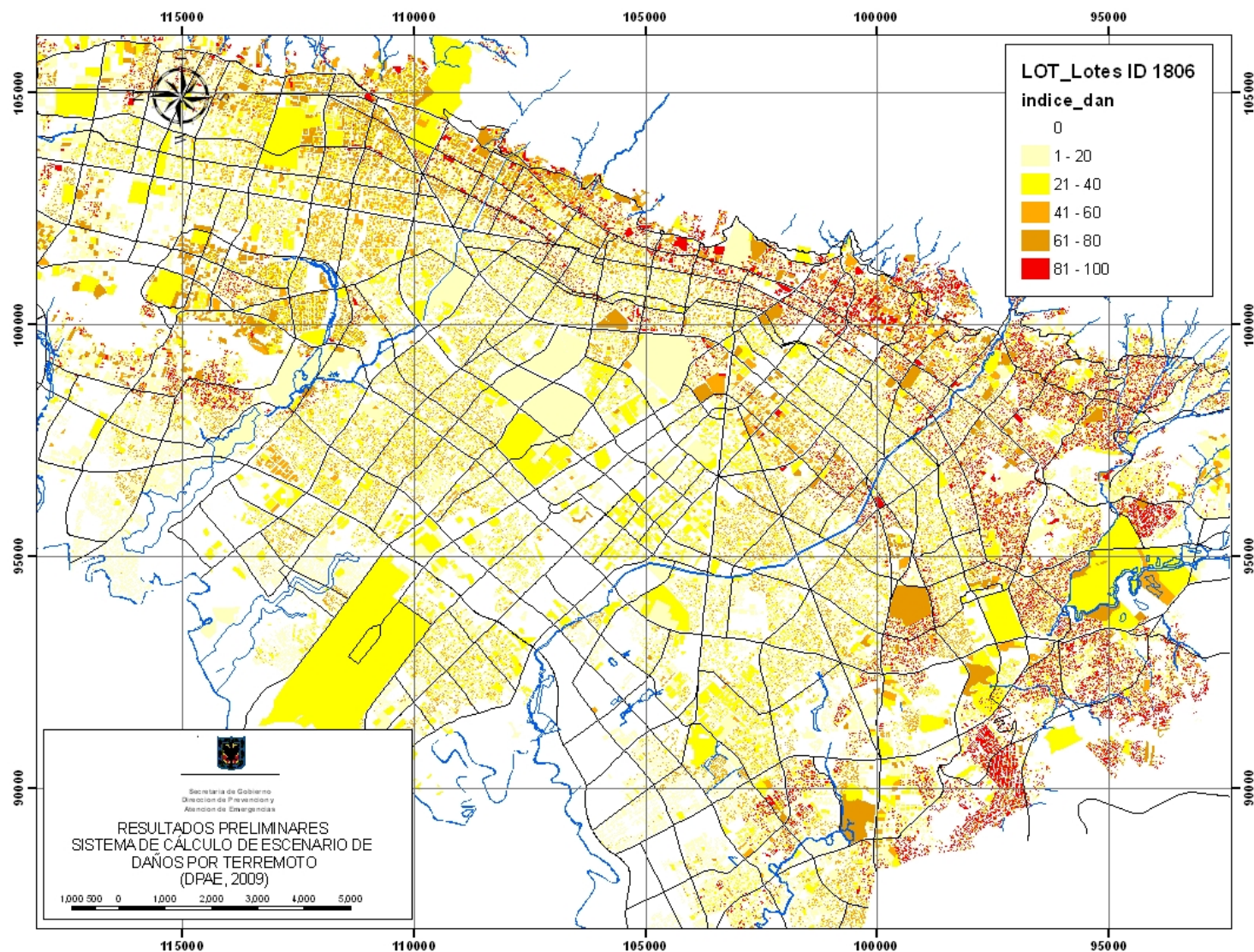


Figura 6.4. Resultados preliminares del prototipo de índice de daño de edificaciones

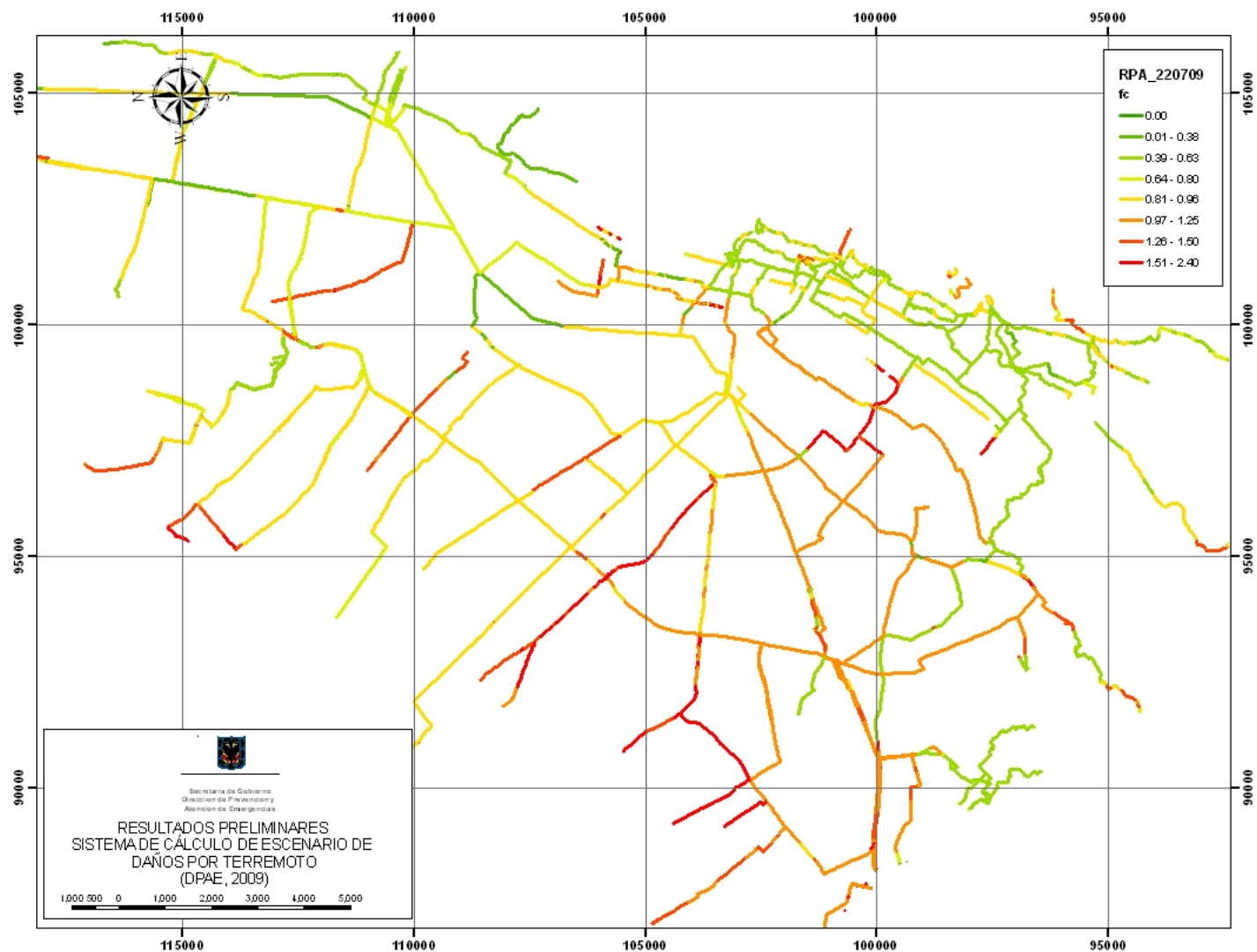


Figura 6.5. Resultados preliminares del prototipo de índice de vulnerabilidad de la red de acueducto matriz

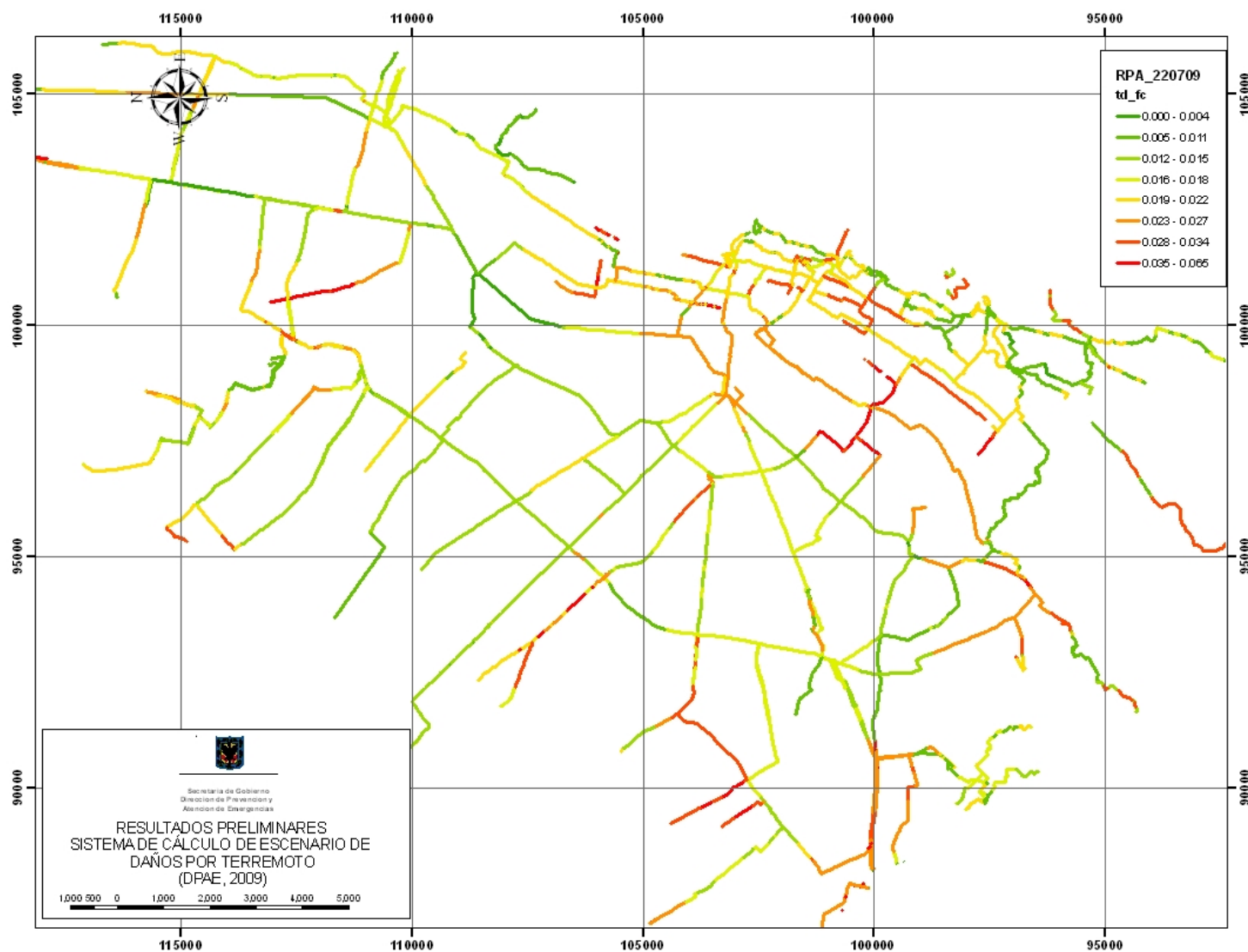


Figura 6.6. Resultados preliminares del prototipo de tasa de roturas de la red de acueducto matriz

7 CONCLUSIONES Y RECOMENDACIONES

Las conclusiones y recomendaciones dadas en el presente informe corresponden al avance y resultados del proyecto en su primera fase, las cuales son considerables y un buen punto de partida base para retroalimentar las siguientes actividades de la siguiente fase del estudio.

- Se ha seleccionado a Java como lenguaje de programación, y se ha definido la plataforma de ejecución de la aplicación. La selección del lenguaje de programación implicó desarrollar una aplicación independiente y no un parche o plug-in que se ejecute dentro de los módulos de Arc-gis, ya que éste último no permite registrar en su interfaz archivos Jar de Java, solo dlls.
- El prototipo del sistema de cálculo de escenario de daños cuenta con la definición de la plataforma tecnológica, la arquitectura del programa, un modelo de datos espaciales, y se ha logrado programar dos módulos de cálculo, índice de vulnerabilidad de edificaciones y tasas de roturas de líneas vitales. Es importante aclarar que a la fecha del presente informe de se cuenta con una versión prototipo del software, su desarrollo aproximadamente equivale a un 50% del trabajo que se plantea en este proyecto, el 50% restante se tiene programado realizar en la segunda fase del estudio.
- Se definió del contenido del sistema, numeración de secciones, módulos, rutinas y subrutinas, avance en el diseño del modelo de datos espacial y base de datos del sistema; creación y cargue de la base de datos del sistema acorde con el avance en el modelo de datos.
- Dada la heterogeneidad de la calidad de la información recopilada se recomienda para las próximas actividades relacionadas realizar una validación más exhaustiva de la información. Así mismo es necesario que desde el grupo de información y sistemas de la entidad se trabaje en la centralización de toda la información geográfica para fines de trabajos que requieren análisis temáticos por los especialistas de la Entidad, en los cuales se requiere del acceso a las bases de datos geográficas y una estructura geomática definida, esto es diferente al sistema actual para consulta que se realiza por Internet o solicitudes de figuras o mapas temáticos.
- Para la siguiente fase se recomienda explorar nuevas formas de programar con los objetos de Arcgis que obtienen información de la base de datos, ya que no se han agotado todas las posibilidades que ofrece la Api de ArcObjects. También es recomendable observar la posibilidad de eliminar el campo "XXX_ID" (donde XXX es el nombre de la tabla) que tienen las tablas del modelo de datos, ya que simplemente son un duplicado del campo objectID que por defecto ofrece Arcgis, y no están ofreciendo ningún beneficio. Por último es aconsejable adelantar pruebas con grandes volúmenes de datos, para verificar que una misma base de datos pueda albergar múltiples resultados (cada resultado implica una gran cantidad de datos en una tabla) sobre una cobertura, y si no es así, buscar la manera de que el software administre estos resultados en diferentes bases de datos sin que el usuario perciba la diferencia.

8 REFERENCIAS

Asociación Colombiana de Ingeniería Sísmica AIS, (1998). Normas Colombianas de Diseño y Construcción Sismo Resistente NSR 98, Ley 400 de 1997, Decreto 33 de 1998.

Cardona, O. D. (2001). Estimación holística del riesgo sísmico utilizando sistemas dinámicos complejos. Universidad Politécnica de Cataluña, PhD thesis, Barcelona, Spain. <http://www.tdcat.cesca.es/TDCat-0416102-075520/> (consultada Julio 2008)

Grady Booch, James Rumbaugh, Ivar Jacobson, (2007). El Lenguaje Unificado de Modelado. Manual de Referencia. PEARSON EDUCATION. España. 638 Paginas.

INGEOMINAS, UNIANDES, (1997). Microzonificación Sísmica de Bogotá. Publicación especial Ingeominas. Bogotá.

Japan International Cooperation Agency-JICA (2002). The Study on Disaster Prevention in the Bogotá Metropolitan Area in the Republic of Colombia. Draft Final Report. Main. January.

Oliveira C.S., Roca A., and Goula X. (2006). Assessing and Managing Earthquake Risk. Springer. Netherlands.

Sierra Katty, Beart Batles. SCJP Sun certified programmer for Java 5 Study Guide. Exam (310-055) McGraw –Hill/Osborne / book – 825

UNIANDES Universidad de los Andes, (2005). Escenarios de Riesgo y Pérdidas por Terremoto para Bogotá, Centro de Estudios e investigaciones sobre Riesgo. Bogotá.

Páginas web

The java Tutorials – Sun Microsystems. Lesson: Exceptions.

<http://java.sun.com/docs/books/tutorial/essential/exceptions/index.html>

Object Management Group (OMG)

www.uml.org SCJP Sun Certified Programmer for Java 5 Study Guide: Exam (310-055)