

## DEFINICIÓN DE UN MÉTODO BASADO EN PATRONES DE ANÁLISIS PARA LA INTEROPERABILIDAD ENTRE SISTEMAS DE INFORMACIÓN GEOGRÁFICA

CARLOS MARIO ZAPATA\*  
FRANCISCO MAURICIO TORO\*\*  
MARÍA ISABEL MARÍN\*\*\*

### RESUMEN

Los Sistemas de Información Geográfica (SIG) se incorporan en las empresas para facilitar el análisis espacial de la información. Este análisis suele requerir la integración de datos residentes en SIG heterogéneos que tienen la necesidad de interoperar. En los SIG, la interoperabilidad se aborda, principalmente, con la especificación de estándares y la implementación de ontologías, servicios web, computación *grid* y aplicaciones intermedias. Para implementar cualquiera de estos mecanismos, siempre es necesario que un analista realice un estudio detallado del dominio SIG para cada situación de interoperabilidad que se presente. Este estudio suele ser subjetivo y dependiente de la experiencia del analista, lo que afecta su transparencia y dificulta su trazabilidad. Por esta razón, en este artículo se propone un método de interoperabilidad que generaliza la fase de análisis del dominio SIG. El método consta de siete pasos que se apoyan en patrones de análisis expresados en esquemas preconceptuales. Además, se puede integrar con los mecanismos de interoperabilidad mencionados, reduciendo los tiempos en la fase de análisis y brindando un canal de comunicación entre el usuario, el analista y el desarrollador.

**PALABRAS CLAVE:** interoperabilidad; Sistemas de Información Geográfica; modelo de datos *vector*; modelo de datos *raster*; series de datos; patrones de análisis.

---

\* Ingeniero Civil, Magíster en Ingeniería de Sistemas, Doctor en Ingeniería con énfasis en Sistemas, Universidad Nacional de Colombia. Profesor Asociado y líder del grupo de investigación en Lenguajes Computacionales, Escuela de Sistemas, Universidad Nacional de Colombia. Medellín, Colombia. cmzapata@unal.edu.co

\*\* Ingeniero Civil, Universidad Nacional de Colombia; Master of Science in Engineering y Ph. D. in Hidrosience and Engineering, University of Mississippi. Profesor Titular, Integrante de los grupos de investigación PARH y Oceánicos, Escuela de Geociencias y Medio Ambiente, Universidad Nacional de Colombia. Medellín, Colombia. fmoro@unal.edu.co

\*\*\* Ingeniera de Sistemas e Informática y Magíster en Ingeniería de Sistemas, Universidad Nacional de Colombia. Medellín, Colombia. mimarinm@unal.edu.co

## DEFINITION OF AN ANALYSIS-PATTERN-BASED METHOD FOR THE INTEROPERABILITY BETWEEN GEOGRAPHIC INFORMATION SYSTEMS

### ABSTRACT

Geographic Information Systems (GIS) are incorporated in the companies to facilitate spatial analysis of the information. This analysis often requires the integration of GIS heterogeneous data to interoperate. In GIS, interoperability is mainly addressed by the specification of standards and the implementation of several mechanisms like ontologies, web services, grid computing, and middleware technologies. In order to implement any of such mechanisms, an analyst should make a detailed and iterative study of the GIS domain for every situation of interoperability is needed. Such study is, commonly, subjective and highly dependent on the analyst experience. Consequently, the entire process seems not to be transparent and traceable. For this reason, in this paper we propose an interoperability method for generalizing the analysis phase of the GIS domain. The method comprises seven steps supported by analysis patterns expressed in pre-conceptual schemas. The method can be integrated with the aforementioned mechanisms for interoperability, thus shortening the analysis phase and providing a communication channel between the user, the analyst, and the developer.

**KEY WORDS:** interoperability; Geographic Information Systems; vector data model; raster data model; data sets; analysis patterns.

## DEFINIÇÃO DE UM MÉTODO BASEADO EM PADRÕES DE ANÁLISES PARA A INTEROPERABILIDADE ENTRE SISTEMAS DE INFORMAÇÃO GEOGRÁFICA

### RESUMO

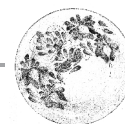
Os Sistemas de Informação Geográfica (SIG) incorporam-se nas empresas para facilitar a análise espacial da informação. Esta análise costuma requerer a integração de dados residentes em SIG heterogêneos que têm a necessidade de interoperar. Nos SIG, a interoperabilidade aborda-se, principalmente, com a especificação de normas e a implementação de ontologias, serviços web, computação grid e aplicações intermediárias. Para implementar qualquer destes mecanismos, sempre é necessário que um analista realize um estudo detalhado do domínio SIG para a cada situação de interoperabilidade que se presente. Este estudo costuma ser subjetivo e dependente da experiência do analista, isso afeta sua transparência e dificulta sua rastreabilidade. Por esta razão, neste artigo propõe-se um método de interoperabilidade que generaliza a fase de análise do domínio SIG. O método consta de sete passos que se apoiam em padrões de análises expressados em esquemas preconceituais. Ademais, pode ser integrado com os mecanismos de interoperabilidade mencionados, reduzindo os tempos na fase de análise e brindando um canal de comunicação entre o usuário, o analista e o desenvolvedor.

**PALAVRAS-CÓDIGO:** interoperabilidade; Sistemas de Informação Geográfica; modelo de dados *vector*; modelo de dados *raster*; séries de dados; padrões de análises.

### 1. INTRODUCCIÓN

Debido a la masificación de los instrumentos de adquisición de información geográfica, se facilitó su uso en empresas y corporaciones. Los radares

y los sensores remotos, entre otros instrumentos, constantemente toman datos de diferentes variables sobre la tierra, la atmósfera y el océano (Rozenstein y Karnieli, 2011). Con estos instrumentos, las empresas de diferentes áreas del conocimiento utilizan la



información georreferenciada en sus negocios para tomar decisiones. La información geográfica se almacena, procesa y visualiza por medio de Sistemas de Información Geográfica (SIG).

El análisis integrado de la información geográfica incorpora diferentes fuentes de información y de SIG heterogéneos entre sí. Para que sea posible la integración entre estos sistemas se necesita que posean la capacidad de interoperar, que se define como la capacidad de dos o más aplicaciones para intercambiar datos y usarlos en funciones y procesos propios (Budiarto, Pradeep y Aziz, 2009).

La interoperabilidad entre SIG se aborda más que todo mediante la especificación de estándares y la implementación de ontologías, servicios web, computación *grid* y aplicaciones intermedias. Organizaciones como el Open Geospatial Consortium (OGC) cumplen un papel protagónico en el proceso de estandarización de formatos de almacenamiento y transporte de los datos y de interfaces para la implementación de servicios web (Rao, Percivall y Enloe, 2000).

Para la implementación de cada uno de los mecanismos de interoperabilidad mencionados, siempre es necesario estudiar el dominio de los SIG en la fase de análisis, especificando los tipos y modelos de datos soportados y las herramientas incorporadas en cada SIG. Este proceso se efectúa cada vez que se requiere la implementación de uno de los mecanismos de interoperabilidad. Por ello, la transparencia del proceso se ve afectada, al igual que la trazabilidad que podría permitir su seguimiento, pues cada analista involucra su propia experiencia en el estudio, pese a que los pasos que realizan serían susceptibles de incorporarse en un método independiente del dominio del SIG y del mecanismo de interoperabilidad. En consecuencia, teniendo en cuenta que las características inherentes al dominio SIG no varían de un proyecto a otro y que se requiere la aplicación sistemática del estudio de interoperabilidad, tomando como base la ingeniería de métodos (Sunyaev, Hansen y Krcmar, 2010), se puede plantear un método que sistematice este proceso.

Por lo anterior, en este artículo se propone un método de interoperabilidad que generaliza la fase de análisis del dominio SIG. El método se compone de siete pasos: a) situación actual de interoperabilidad, donde se identifica el estado de interoperabilidad de los sistemas involucrados; b) identificación de las fuentes de datos soportadas; c) definición de los formatos aceptados; d) intercambio de datos por medio de una API; e) identificación de la estructura de las fuentes de datos, que se basa en patrones de análisis expresados en esquemas preconceptuales; f) identificación de las operaciones soportadas; g) operación paso a paso, donde se realizan los pasos recomendados para llevar a cabo cada una de estas operaciones. El método se puede integrar con los mecanismos de interoperabilidad mencionados, reduciendo los tiempos en la fase de análisis y brindando un canal de comunicación entre el usuario, el analista y el desarrollador.

Este artículo se estructura así: en la sección 2 se expone el marco teórico que acerca al lector a los conceptos de SIG, interoperabilidad y patrones de análisis; en la sección 3 se hace una recopilación de los avances alcanzados en los patrones de análisis y en la interoperabilidad entre SIG; en la sección 4 se presenta el método propuesto; en la sección 5 se expone un ejemplo de interoperabilidad usando el método; finalmente, en la sección 6 se sintetizan las conclusiones y se plantea el trabajo futuro que se puede derivar del actual.

## 2. MARCO TEÓRICO

### 2.1 Sistemas de Información Geográfica

Son aplicaciones que poseen datos ligados con unas coordenadas geográficas y que se estructuran en capas temáticas independientes para representar diferentes variables. Los SIG se diseñan para captar, almacenar, procesar y desplegar información georreferenciada. Estos sistemas se emplean para

investigaciones científicas, gestión de los recursos, evaluación del impacto ambiental, planificación urbana, cartografía, sociología y mercadeo, entre otros campos.

Cada capa de un SIG corresponde a uno de estos tipos de datos: los modelos de datos *raster* y *vector* (Bolstad, 2005). Además, en ocasiones se asocian series de datos a estos modelos. Estos tipos de datos se detallan a continuación.

El modelo de datos *raster* permite dividir el área de estudio en una matriz de datos. Cada dato se asocia con una celda o píxel de la imagen desplegada en el SIG cuyo valor se considera representativo para toda la superficie que abarca la celda. El modelo *raster* representa la información continua en el espacio, midiendo variables como la elevación y la precipitación, entre otras (DeMers, 2002).

En el modelo *vector*, el interés de la representación se centra en la precisión de la localización y la forma. Este modelo permite representar la información por medio de primitivas geométricas, como los puntos, las líneas y los polígonos. Cada primitiva se vincula con una fila en una base de datos que describe sus atributos. El modelo de datos *vector* representa la información discreta en el espacio, estableciendo elementos como casas, calles, ciudades, etc. (De Smith, Goodchild y Longley, 2007).

Las series de datos corresponden por tradición a series de tiempo o a perfiles de medición (p. ej. verticales y longitudinales). Una serie de datos se puede asociar a un elemento geométrico de un modelo *vector* o a una región de la malla de datos del modelo *raster*. De esta manera, la serie de datos describe la región geográfica con la que se asocia, de acuerdo con la variable que representa: temperatura, velocidad del viento, conductividad eléctrica, etc.

## 2.2 Interoperabilidad entre SIG

De acuerdo con los términos fundamentales del vocabulario de tecnologías de la información de

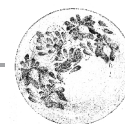
la norma ISO/IEC 2382-01, la interoperabilidad es la capacidad de diferentes unidades funcionales para comunicarse, ejecutar programas o transferir datos, de tal manera que el usuario tenga poco o ningún conocimiento de las características únicas de dichas unidades (Chen *et al.*, 2010).

Zhang *et al.* (2007) definen la interoperabilidad entre SIG como la capacidad de estos sistemas para: 1) intercambiar todo tipo de información espacial sobre los objetos y fenómenos que se dan dentro y sobre la tierra, y 2) ejecutar, cooperativamente sobre la red, unidades funcionales capaces de manipular esta información.

## 2.3 Patrones de análisis

Martin Fowler introdujo este término para describir soluciones relacionadas con problemas que aparecen durante el análisis de requisitos y las fases de modelado conceptual de los datos. Para él “un patrón es una idea que fue útil en un contexto práctico y probablemente será útil en otros” (Fowler, 1997). La semántica de los patrones de análisis describe aspectos específicos de algún dominio de aplicación en concreto, permitiendo definir un conjunto de modelos conceptuales que juntos describen un lenguaje del dominio (Fowler, 1997).

En la ingeniería de software, los patrones se usan para permitir la reusabilidad de soluciones exitosas de problemas recurrentes. El propósito de los patrones se centra en incorporar calidad y productividad en las fases de desarrollo del software y brindar al desarrollador un punto de partida. La especificación de los patrones de análisis se puede realizar por medio de narración libre, plantillas o modelos conceptuales. Es necesario que el mecanismo de representación cumpla con los siguientes requisitos: a) ser fácil de comprender para usuarios no expertos en el dominio; b) tener una alta capacidad de expresión; y c) ser extensible (Lisboa, Iochpe y Borges, 2002).

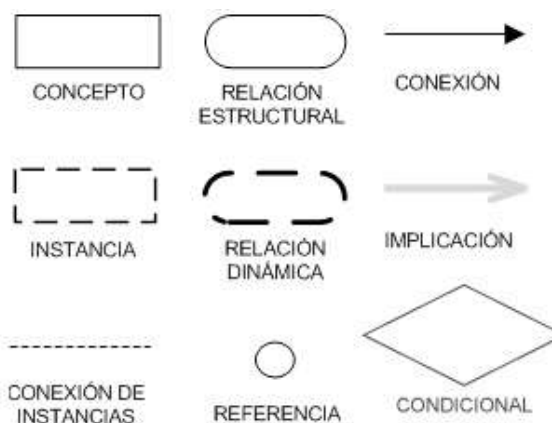


## 2.4. Esquemas preconceptuales

Los esquemas preconceptuales son representaciones intermedias entre las especificaciones textuales en lenguaje natural de cualquier dominio del conocimiento y los diferentes esquemas conceptuales que permiten el modelado de una aplicación de software. Estos esquemas permiten representar las relaciones estructurales y dinámicas del dominio (Zapata, Gelbukh y Arango, 2006). En la figura 1 se presentan los elementos de los esquemas preconceptuales que se utilizarán en este artículo y cuyos significados se explican en la tabla 1.

**Tabla 1.** Algunos elementos de los esquemas preconceptuales. Adaptado de Zapata, Gelbukh y Arango (2006) y de Zapata, Tamayo y Arango (2007)

| Elemento                 | Explicación   |
|--------------------------|---|
| Conceptos                | Representan los sustantivos del dominio   |
| Instancias               | Son conjuntos de valores que puede tomar un concepto y que sirven para aclararlo  |
| Relaciones estructurales | Definen relaciones permanentes entre dos conceptos usando únicamente los verbos “ser” y “tener”   |
| Relaciones dinámicas     | Son verbos que denotan acciones u operaciones que generan relaciones transitorias entre los conceptos (v. g., registra, paga, asigna, etc.) |
| Referencias              | Son elementos que permiten ligar conexiones desde o hacia conceptos físicamente distantes en el esquema preconceptual                       |
| Conexiones               | Son vínculos entre conceptos y relaciones (dinámicas o estructurales) y viceversa   |
| Implicaciones            | Sirven para unir relaciones dinámicas, estableciendo entre ellas una relación de causa-efecto   |
| Conexiones de instancias | Permiten ligar un conjunto de instancias con el concepto del cual son valores   |
| Condicionales            | Son relaciones de causalidad que indican las condiciones o reglas del negocio que se deben cumplir  |



**Figura 1.** Algunos elementos de los esquemas preconceptuales. Fuente: Zapata, Gelbukh y Arango (2006), Tamayo y Arango (2007)

## 3. ANTECEDENTES

### 3.1 Interoperabilidad entre Sistemas de Información Geográfica

La interoperabilidad entre SIG se aborda, principalmente, con el uso de varios mecanismos: la especificación de estándares y la implementación de servicios web, la computación *grid* y los sistemas intermedios (*middleware*).

Respecto de la especificación de estándares, el OGC es la organización que vincula el mayor número de integrantes de empresas privadas, agencias gubernamentales y universidades en el proceso de consenso (Rao, Percivall y Enloe, 2000). Los estándares se dividen en dos grupos: para el almacenamiento y transporte de datos y para la implementación de servicios web. Los principales estándares para el almacenamiento y transporte de los datos geográficos son: *Geography Markup Language* (GML), *Keyhole Markup Language* (KML) y *Network Common Data Form* (NetCDF). El estándar GML es una gramática XML que emplea otras tecnologías y estándares para especificar estilos de visualización (p. ej. SVG, HTML) del contenido; GML no se limita a una aplicación

de visualización específica, pues se puede transformar con hojas de estilo para su visualización en *Google Earth* (Sripada, Lu y Wu, 2004). El estándar KML es un lenguaje de marcado para representar datos geográficos en tres dimensiones, basado en GML (Du, Yu y Liu, 2009). El estándar NetCDF es un formato binario que incorpora librerías para su procesamiento en algunos lenguajes como Java y MATLAB, almacenando paralelepípedos rectangulares de datos, en donde la base se constituye con las variables espaciales (por ejemplo, latitud y longitud) y la altura representa la variable tiempo. Además permite describir los datos con atributos y anotaciones (Rew y Davis, 1990).

En cuanto a la especificación de servicios web, el OGC definió los servicios *Web Feature Service* (WFS), *Web Coverage Service* (WCS), *Web Map Service* (WMS) y *Catalogue Service for Web* (CSW). El servicio WFS permite recuperar y consultar modelos de datos *vector* en la web, proporcionando operaciones como *GetCapabilities*, *DescribeFeatureType* y *GetFeature*; WCS brinda un medio para la recuperación de capas *raster* en la web y define las operaciones *GetCapabilities*, *DescribeCoverage* y *GetCoverage*; WMS posibilita la recuperación de mapas (imágenes) en internet, ofreciendo las operaciones *GetCapabilities*, *GetMap* y *GetFeatureInfo*. El servicio CSW presenta una interfaz para el descubrimiento de servicios geográficos en la web (Fan, Feng y Zhu, 2009).

El *Open-source Project for a Network Data Access Protocol* (OPeNDAP) también definió una especificación para servicios web que lleva el mismo nombre. Se basa en HTTP y define una arquitectura de transporte de datos para encapsular la estructura de la información. El servicio devuelve datos en el formato NetCDF (Xu, Li y Mi, 2010).

A más del uso de estándares y servicios web, la implementación de ontologías y de computación *grid* también permite la interoperabilidad entre SIG. Con las ontologías, el problema se aborda desde el nivel semántico. En esta dirección se construyen taxonomías y ontologías en el dominio (Yan, Li y He,

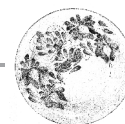
2006). Por otro lado, la computación *grid* permite trabajar con información geográfica distribuida en la red y se apoya en estándares para compartir los datos y procesos mediante servicios web (Al-Hader *et al.*, 2009).

Respecto de los sistemas intermedios, existen iniciativas como las de McDonald *et al.* (2006) y Hu *et al.* (2008) para posibilitar la interoperabilidad entre sistemas que incorporan diferentes estándares. Ellos implementan un *gateway* por cada dirección de solicitud de los servicios, es decir, un *gateway* para permitir la interoperabilidad entre clientes que implementan protocolos OPeNDAP y servidores que implementan protocolos OGC y otro para que clientes OGC puedan acceder a los servicios de un servidor que implementa protocolos OPeNDAP.

La implementación de cualquiera de los mecanismos anotados para facilitar la interoperabilidad entre SIG requiere un estudio detallado que incluya las características principales de cada uno de los SIG que participan en el proceso de interoperabilidad. En este estudio se involucran, normalmente, la subjetividad y la experiencia del analista, quien se encarga de posibilitar la interoperabilidad entre los SIG. Así, pese a que los pasos generales pueden ser comunes y cada analista los realiza a su manera, generando problemas de transparencia en el manejo de la información y trazabilidad de las características del proceso mismo. Estos problemas constituyen el fundamento para que Sunyaev, Hansen y Krmar (2010) sugieran el planteamiento de un método que contribuya a aminorar estos problemas. La definición del método que incluye la ejecución sistemática de estos pasos constituye el fundamento del presente artículo.

### 3.2 Patrones de análisis en Sistemas de Información Geográfica

Los patrones de análisis en los SIG se desarrollan para casos de aplicación específicos. En este numeral se exponen algunos casos relevantes.



Lisboa, Iochpe y Beard (1998) derivan y aplican patrones de análisis en el proceso de modelado conceptual de datos de aplicaciones geográficas. Estos autores introducen el patrón “parámetros de calidad ambiental” (*environmental quality parameters*). En este patrón se puede identificar la relación entre puntos de muestreos y las mediciones. Adicionalmente, los puntos de muestreos son recursos hídricos que pueden ser lagos o ríos, cada uno de los cuales se asocia con una cuenca.

Iochpe y Lisboa (1998) proponen una jerarquía básica de clases para apoyar el análisis conceptual de los SIG. La jerarquía se divide en dos niveles: conceptual y de representación espacial. En el primero se especifican con mayor detalle los grupos del nivel conceptual. El segundo permite identificar las agrupaciones conceptuales dentro de los SIG, como los objetos geográficos y los campos geográficos, entre otros.

Lisboa e Iochpe (1999) proponen *GeoFrame*, un marco conceptual que proporciona un diagrama básico de clases para ayudar al diseñador en los primeros pasos del modelado conceptual de datos de una nueva aplicación SIG.

Lisboa, Iochpe y Borges (2002) aplican los patrones de análisis para la conceptualización del conjunto de sistemas de gestión urbana (por ejemplo, sistemas de control tributario y sistema de transporte urbano). Este trabajo propone tres patrones de análisis que hacen posible la reutilización del diseño de bases de datos geográficas para la planificación del área urbana y las aplicaciones de gestión. Los patrones que proponen son: “malla vial urbana”, “red de tráfico urbano” y “zonificación urbana”.

En esta revisión de antecedentes se identificó que los patrones de análisis en la actualidad se desarrollan en aplicaciones ligadas con dominios específicos y para situaciones concretas, lo que limita su capacidad de reutilización para analizar problemas generales de los SIG. Es necesario abstraerse de la

aplicación del SIG y diseñar patrones de análisis para el dominio SIG como tal (Hamza y Fayad, 2006). Se requiere que estos patrones puedan definirse en un lenguaje que posibilite la estructuración de las características de los SIG involucrados en el proceso de interoperabilidad. Además, sería deseable que se pudieran involucrar también en el desarrollo de futuras aplicaciones SIG y que definieran el proceso por seguir para cada uno de los pasos en que se involucren. Estas características las cumplen los esquemas preconceptuales, que tienen la posibilidad de generar varios esquemas conceptuales para el desarrollo de software y constituyen una forma de representación del conocimiento en un dominio determinado (Zapata, Gelbukh y Arango, 2006).

## 4. MÉTODO PROPUESTO

Cuando se acomete la implantación de un mecanismo que posibilite la interoperabilidad entre SIG, la fase de análisis posee un conjunto de tareas. El método de interoperabilidad que se propone a continuación contiene la aplicación sistemática de una serie de pasos que permiten al analista conocer los elementos que debe identificar en cada SIG involucrado. Estos elementos se definen en términos de los tipos de datos SIG y las operaciones características de este dominio. El método consta de siete fases secuenciales que el analista debe llevar a cabo y que se explican enseguida.

### 4.1 Situación actual de interoperabilidad

Se busca en este paso establecer la situación actual de interoperabilidad para ayudar a identificar el nivel de heterogeneidad de los sistemas y, de esta manera, elegir un mecanismo específico y el esfuerzo que representará su desarrollo. Para ello, se definen los cinco criterios que se especifican en la tabla 2, que se deben identificar para cada sistema que se interopera.

**Tabla 2.** Criterios de situación actual de interoperabilidad

| Criterio                                 | Explicación   |
|--|---|
| Tipo de aplicación                       | Los tipos de aplicaciones son: a) de escritorio, b) web y c) híbrido  |
| Framework de desarrollo                  | El uso de <i>frameworks</i> de desarrollo distantes entre aplicaciones dificulta el proceso de interoperabilidad  |
| Estándares implementados                 | Si las aplicaciones implementan estándares en común, la interoperabilidad se da directamente  |
| Interfaz de programación de aplicaciones | La existencia de una API en un SIG permite que los programas puedan acceder por su intermedio a los modelos de datos y no directamente a las fuentes de datos |
| Sistema operativo que lo soporta         | Superar heterogeneidades en términos de sistemas operativos diferentes requiere mayor esfuerzo en el proceso de interoperabilidad                             |

### 4.2 Identificación de las fuentes de datos soportadas

En el proceso de conceptualización de los SIG para interoperar, es necesario identificar las diferentes fuentes de datos que soportan los SIG que intervienen en el proceso. Para ello, se definieron previamente las diferentes opciones que se presentan en la tabla 3.

**Tabla 3.** Fuentes de datos soportadas

| Fuente de datos               | Explicación   |
|-------------------------------|---|
| PostGIS                       | Presenta una estructura definida y conocida para el acceso a los datos, ya que cumple con los estándares que establece el OGC. Es una base de datos centralizada que corre sobre el sistema de gestión de bases de datos PostgreSQL |
| Servidores de mapas           | Presentan una estructura definida y conocida para el acceso a los datos, ya que cumplen con los estándares que establece el OGC. Presentan los servicios empleando internet   |
| Bases de datos especializadas | Son aquellas que se crean en cada SIG para proyectos específicos y cuya estructura no es pública, incorporando un nivel adicional en el estudio específico de cada SIG  |
| Archivos en disco             | Son muchos los formatos de archivos que se pueden abrir directamente desde el disco duro. Es necesario conocer los diferentes formatos aceptados  |

### 4.3 Definición de los formatos aceptados

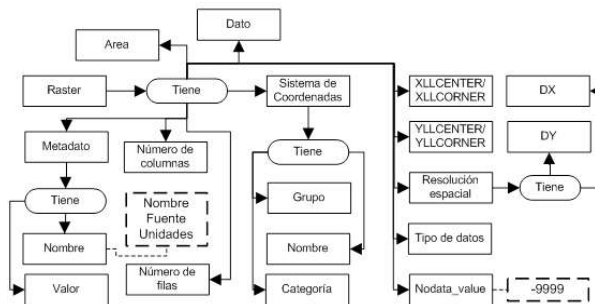
Si dentro de las fuentes de datos soportadas se incluyeron los archivos en disco, es necesario especificar sus formatos. En la tabla 4 se presenta una lista de posibles formatos soportados, dependiendo de los modelos de datos que el analista puede tener en cuenta a la hora de explorar cada SIG.

**Tabla 4.** Formatos y modelos de datos

| Modelo de datos | Formato  |
|-----------------|--|
| Raster          | asc, img, ovr, lgg, bag, bil, bip, bsq, trl, gis, grd, hdf, png, tif, bmp, dat, gif, jpg, grid |
| Vector          | shp, dbf, shx  |
| Series de datos | NetCDF, HDF, CDF   |

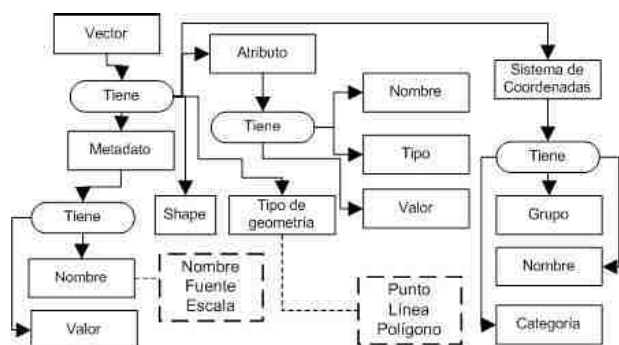
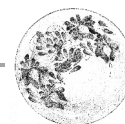
### 4.4 Intercambio de datos por medio de una API

Si el SIG cuenta con una API, es necesario conocer la abstracción que esta implementa para recibir y entregar los datos. Por lo tanto, el analista debe identificar en la API cada concepto presente en los patrones de análisis sobre los tipos de datos geográficos identificados: *raster*, *vector* y series de datos. Los patrones de análisis se proponen haciendo uso de los esquemas preconceptuales que se presentan en las figuras 2, 3 y 4 para los modelos *raster*, *vector* y las series de datos, respectivamente.

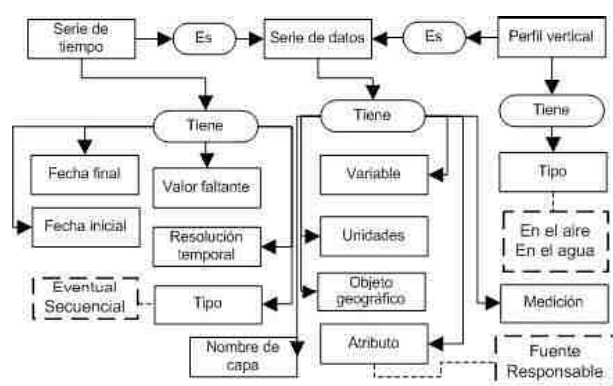


**Figura 2.** Patrón de análisis para el modelo de datos *raster*





**Figura 3.** Patrón de análisis para el modelo de datos *vector*



**Figura 4.** Patrón de análisis para las series de datos

#### 4.5 Identificación de la estructura de las fuentes de datos

Si el SIG no cuenta con una API, quiere decir que para realizar la interoperabilidad es necesario acceder directamente a las fuentes de datos y determinar para cada una su estructura. En este caso, el analista debe identificar en cada fuente de datos cada concepto presente en los patrones de análisis sobre los tipos de datos geográficos expuestos en las figuras 2, 3 y 4.

#### 4.6 Identificación de las operaciones soportadas

En este paso el analista debe identificar las operaciones SIG que soporta cada sistema. A continuación se presenta una lista que sirve de guía para

identificarlas: a) asignar sistemas de coordenadas a elementos geográficos, b) re proyectar elementos geográficos, c) convertir modelos de datos, d) remuestrear modelos *raster*, e) complementar modelos *raster*, f) complementar modelos *vector*, g) cortar modelo *raster* a partir de un polígono, h) cortar modelo *vector* a partir de un polígono, i) complementar atributos de una capa *vector* con los datos de una capa *raster*, j) asociar series de datos a objetos geográficos y k) complementar series de tiempo.

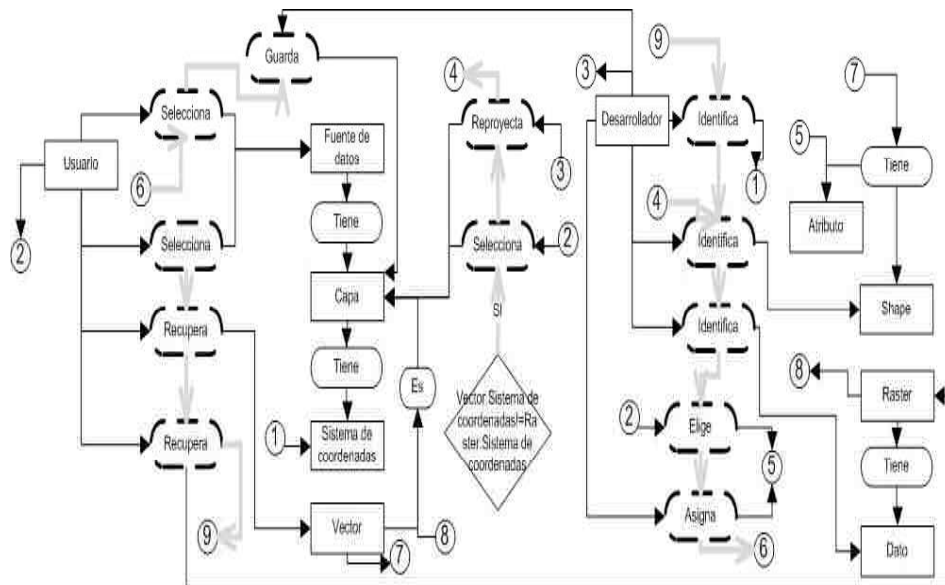
De cada operación, se debe reportar el nombre asignado en cada SIG y tipos de entrada y salida: <nombre de la operación>: tipo(s) de entrada separados por coma -> tipo(s) de salida separados por coma. Las opciones de tipos de datos son *raster*, *vector* y serie de datos.

#### 4.7 Operación paso a paso

Cuando en el proceso de interoperabilidad se seleccione una de las operaciones expuestas en el paso anterior, para su ejecución se deben seguir las recomendaciones paso a paso con base en los patrones de análisis sobre las operaciones típicas de los SIG. En la figura 5 se presenta el patrón de análisis para la operación “Complementar atributos de una capa *vector* con los datos de una capa *raster*”. Esta operación se presenta en la Sección 5.

Con estos siete pasos secuenciales se representa el dominio de los SIG en la primera fase de desarrollo del software en términos de las operaciones y los tipos de datos. Adicionalmente, se hace una exploración de la situación de interoperabilidad entre los sistemas que puede guiar la elección de un mecanismo de interoperabilidad apropiado. Este proceso arroja un importante insumo para las fases siguientes en el proceso de implementación de un mecanismo de interoperabilidad.

Finalmente, los patrones de análisis facilitan la comunicación con el interesado y el desarrollador e incluso facilitan el proceso de levantamiento de requisitos de un proyecto de desarrollo de un nuevo SIG independiente del dominio de aplicación.



**Figura 5.** Patrón de análisis para la operación “Complementar atributos de una capa *vector* con los datos de una capa *raster*”.

## 5 CASO DE ESTUDIO

Este caso de estudio se efectuó entre los SIG GeoDA 0.9 y GvSIG 1.11. El objetivo para la interoperabilidad entre los dos es hacer la operación “Complementar atributos de una capa *vector* con los datos de una capa *raster*”. La capa *raster* se gestiona con el SIG GvSIG desde archivos en disco mientras que la capa *vector* se gestiona con el SIG GeoDA desde un archivo *shapefile*. Los datos usados se to-

maron de la base de datos del proyecto “Estudio de la problemática ambiental de tres embalses de EPM” que se ejecuta en la Escuela de Geociencias y Medio Ambiente de la Universidad Nacional de Colombia, Sede Medellín, en asocio con la Universidad de Antioquia y las Empresas Públicas de Medellín. Los pasos para llevar a cabo el proceso de interoperabilidad se detallaron en la sección 4. Enseguida se exponen los resultados obtenidos (tablas 5, 6 y 7).

### 5.1 Situación actual de interoperabilidad

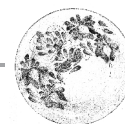
**Tabla 5.** Situación actual de interoperabilidad GeoDA y GvSIG

| SIG   | Tipo de aplicación | Framework de desarrollo | Estándares implementados | API | Sistema operativo      |
|-------|--------------------|-------------------------|--------------------------|-----|------------------------|
| GeoDA | Escritorio         | C++                     | ESRI                     | No  | Windows, Mac OS, Linux |
| GvSIG | Escritorio         | Java                    | OGC                      | Sí  | Multiplataforma        |

### 5.2 Identificación de las fuentes de datos soportadas

**Tabla 6.** Fuentes de datos soportadas GeoDA y GvSIG

| SIG   | PostGIS | Servidor de mapas | Bases de datos especializadas | Archivos en disco |
|-------|---------|-------------------|-------------------------------|-------------------|
| GeoDA | No      | No                | Sí                            | Sí                |
| GvSIG | Sí      | Sí                | Sí                            | Sí                |



### 5.3 Definición de los formatos aceptados

Tabla 7. Formatos aceptados GeoDA y GvSIG

| GvSIG                        |          | GeoDA        |        |
|------------------------------|----------|--------------|--------|
| Raster                       | Vector   | Raster       | Vector |
| asc, txt, ovr, lgg, bag, bil | shp, dbf | No soportado | shp    |
| bip, bsq, trl, gis, grd, hdf | shx      |              | dbf    |

### 5.4 Intercambio de datos por medio de una API

Este paso no es necesario, porque no se cuenta con API en los SIG seleccionados.

### 5.5 Identificación de la estructura de las fuentes de datos

En este punto y con base en los patrones de análisis definidos, se determinó la estructura de las fuentes de datos involucradas en el proceso de interoperabilidad. A continuación se muestra en detalle el caso concreto para las fuentes de datos utilizadas.

GeoDA (archivo en disco, del tipo shp)

Capa *raster*. Sin soporte, ya que el formato en disco *shp* gestiona solo modelos de datos *vector*.

Capa *vector*. Nombre de la capa: se obtiene del nombre del archivo o la leyenda del mapa en el SIG; sistema de coordenadas: se obtiene del archivo prj que acompaña el archivo shp; metadatos: sin soporte; tipo de geometría: se consigue del ícono de la leyenda del mapa; atributos (campos de tabla de atributos): resultan de la tabla dbf que acompaña al archivo shp; *shape* (campo identificador de *shape*): es el campo *ID* de la tabla de atributos.

GvSIG (archivo en disco, del tipo ESRI ASCII)

Capa *vector*. Sin soporte, ya que el formato en disco ESRI ASCII solo gestiona modelos de datos *raster*.

Capa *raster*. Nombre de la capa: se obtiene del nombre del archivo o la leyenda del mapa en el SIG; sistema de coordenadas: se obtiene del archivo prj que acompaña el archivo de la interfaz de ingreso; metadatos: sin soporte; resolución espacial (DX, DY): se obtiene directamente del archivo, las etiquetas son DX y DY respectivamente o CELLSIZE, en cuyo caso indicaría que se trata de celdas cuadradas; punto de inicio de trazado: se obtiene directamente del archivo; las etiquetas son XLLCORNER o XLLCENTER y YLLCORNER o YLLCENTER; número de filas y columnas: se consigue en forma directa del archivo, son las etiquetas NROWS y NCOLS, respectivamente. Valor faltante: se obtiene en forma directa del archivo, la etiqueta es NODATA\_VALUE.

### 5.6 Identificación de las operaciones soportadas

GvSIG

<Asignar sistemas de coordenadas a elementos geográficos>: raster/vector ->raster/vector

<Reproyectar elementos geográficos>: raster/vector ->raster/vector

<Rasterizar>: vector->raster

<Vectorizar>: raster->vector

<Remuestrear modelos raster>: raster->raster

<Complementar modelos raster>: raster, raster->raster

<Complementar modelos vector>: vector, vector->vector

<Cortar modelo raster a partir de un polígono>: raster, vector->raster

<Cortar modelo vector a partir de un polígono>: vector, vector->vector

<Complementar atributos de una capa vector con los datos de una capa raster>: raster, vector->vector

## GeoDA

<Complementar modelos vector>: vector, vector->vector

<Asignar sistemas de coordenadas a elementos geográficos>: raster/vector ->raster/vector

<Reproyectar elementos geográficos>: raster/vector ->raster/vector

### 5.7 Operación paso a paso

Los anteriores pasos son independientes de los modelos de datos con los que trabaja y de las

operaciones que se van a ejecutar. En este punto, los pasos se especifican en la operación: “Complementar atributos de una capa *vector* con los datos de una capa *raster*”, con base en el patrón de análisis expuesto en la figura 5.

**Paso 1.** El usuario selecciona la(s) fuente(s) de datos que almacenan las capas por complementar. En este caso, las fuentes seleccionadas son “Archivos en disco”, para los dos SIG.

**Paso 2.** El usuario recupera la capa *vector* que desea complementar. Por medio del SIG GeoDA recupera la capa “Estaciones Fijas” que quiera complementar (véase la figura 6).

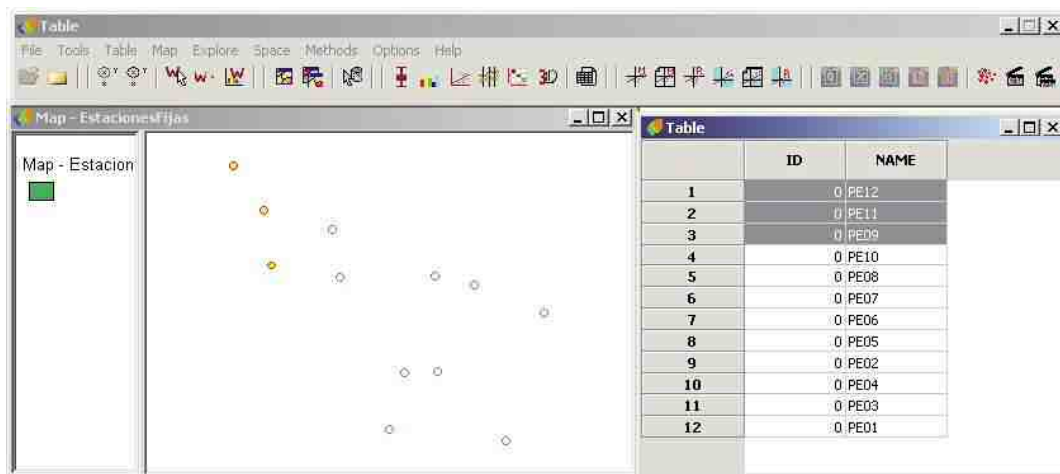


Figura 6. Mapa de estaciones fijas en GeoDA

**Paso 3.** El usuario recupera la capa *raster* de referencia. Por medio del SIG GvSIG se obtiene la capa *raster* de referencia “PORCEII” (véase la figura 7).

**Paso 4.** El programa identifica el sistema de coordenadas de ambas capas, que se puede obtener ya sea de la fuente de datos o directamente de las interfaces gráficas del SIG. Los resultados de este paso se enumeran enseguida. Capa: PORCEII; Sistema de Coordenadas: EPSG 23030 (WGS84) (véase la figura 7); Capa: Estaciones Fijas; Sistema de Coordenadas: Planas Colombia Bogota Zone (véase la figura 8).

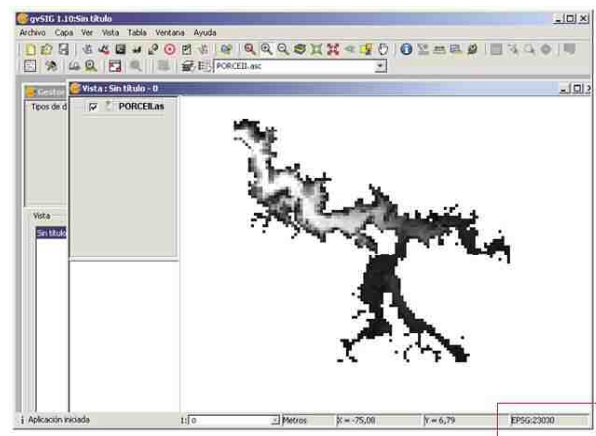
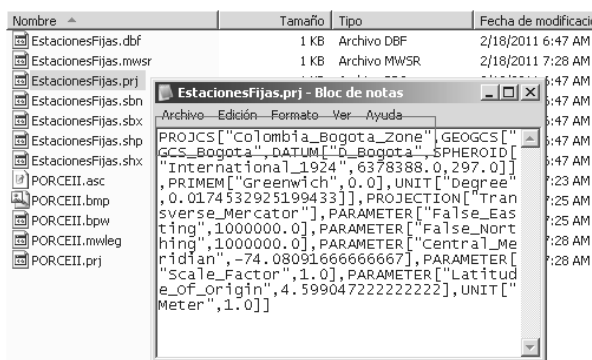
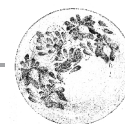


Figura 7. Visualización de sistemas de coordenadas en GvSIG



**Figura 8.** Visualización del sistema de coordenadas desde el archivo prj

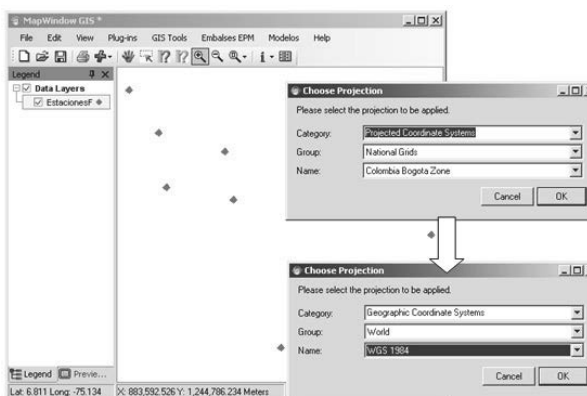
**Paso 5.** Si los sistemas de coordenadas son diferentes, el usuario elige un mapa para reproyectar. En este caso, las dos capas no se hallan en la misma proyección, ya que una está en WGS84 y la otra en Colombia Bogota Zone. Para este caso, es indiferente proyectar una u otra. De esta manera, se selecciona la capa *vector* cuya proyección es Colombia Bogota Zone.

**Paso 6.** Si los sistemas de coordenadas son diferentes, el programa reproyecta el mapa seleccionado. Como se indicó en el paso anterior, se seleccionó la capa *vector* “Estaciones Fijas” para reproyectar. Para hacer este paso se usó la herramienta de reproyección del SIG MapWindow (Aburizaiza y Ames, 2009; véase la figura 9).

**Paso 7.** El programa determina correspondencia entre geometrías y celdas. Este paso se puede efectuar de la manera que cada desarrollador considere más conveniente y, en este caso, indicando las correspondencias uno a uno entre los mapas superponiendo uno sobre el otro. El resultado se muestra en la tabla 8.

**Paso 8.** El usuario elige el campo de la tabla de atributos para asignar los valores. En este caso se eligió el campo MSNM, como se aprecia en el óvalo en la figura 10.

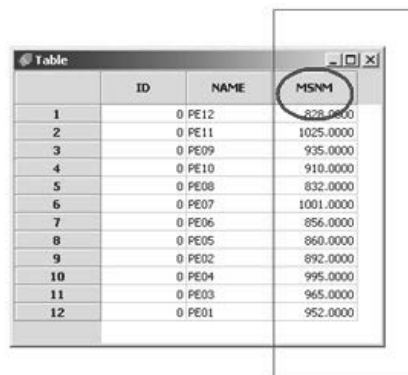
**Paso 9.** El programa almacena los valores en la tabla de atributos. Los datos se ingresaron por medio del SIG GeoDA en el campo MSNM de la tabla de atributos, como se aprecia en la figura 10.



**Figura 9.** Reproyección de mapas con MapWindow

**Tabla 8.** Correspondencias entre geometrías y celdas

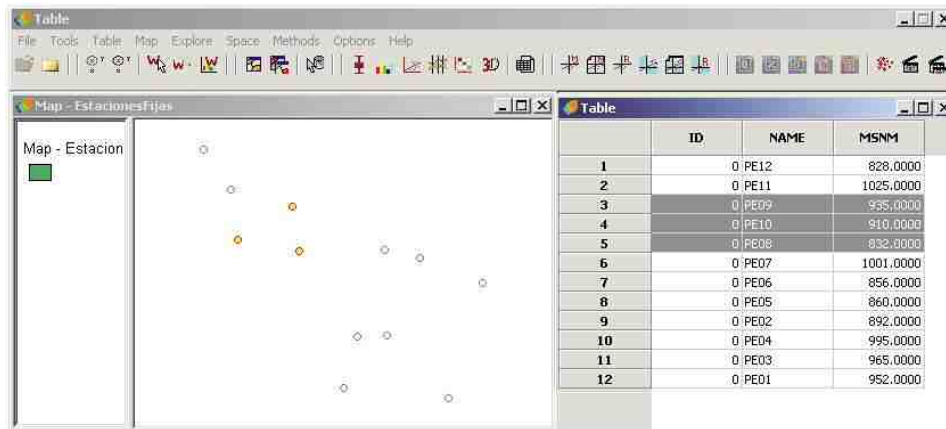
| Shape | Valor      | Shape | Valor    |
|-------|------------|-------|----------|
| 0     | 828 msnm   | 6     | 856 msnm |
| 1     | 1.025 msnm | 7     | 860 msnm |
| 2     | 935 msnm   | 8     | 892 msnm |
| 3     | 910 msnm   | 9     | 995 msnm |
| 4     | 832 msnm   | 10    | 965 msnm |
| 5     | 1.001 msnm | 11    | 952 msnm |



**Figura 10.** Inserción de datos en tabla de atributos

**Paso 10.** El usuario selecciona la fuente de datos donde desea almacenar la capa *vector* resultante. En este caso la fuente de datos seleccionada fue GeoDA.

**Paso 11.** El programa guarda la capa *vector* resultante en la fuente de datos seleccionada. En la figura 11 se muestra la capa resultante con la tabla de atributos.



|    | ID     | NAME | MSNM      |
|----|--------|------|-----------|
| 1  | 0 PE12 |      | 828.0000  |
| 2  | 0 PE11 |      | 1025.0000 |
| 3  | 0 PE09 |      | 935.0000  |
| 4  | 0 PE10 |      | 910.0000  |
| 5  | 0 PE08 |      | 832.0000  |
| 6  | 0 PE07 |      | 1001.0000 |
| 7  | 0 PE06 |      | 856.0000  |
| 8  | 0 PE05 |      | 860.0000  |
| 9  | 0 PE02 |      | 892.0000  |
| 10 | 0 PE04 |      | 995.0000  |
| 11 | 0 PE03 |      | 965.0000  |
| 12 | 0 PE01 |      | 952.0000  |

Figura 11. Capa vector resultante almacenada en GeoDA

## 6 CONCLUSIONES Y TRABAJO FUTURO

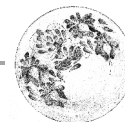
La interoperabilidad es una necesidad que se presenta en forma global y en diferentes dominios del conocimiento. En especial, la interoperabilidad entre SIG se requiere, dada la masificación de los medios de adquisición de la información georreferenciada y la creciente necesidad de integrar sistemas para una mejor toma de decisiones.

Los diferentes mecanismos de interoperabilidad que se listan en la literatura para los SIG (la definición de estándares, el desarrollo de ontologías y la implementación de servicios web, la computación *grid* y los sistemas intermedios) presentan en común en la fase de análisis el estudio detallado del dominio de los SIG. Sin embargo, los analistas encargados del proceso suelen generar problemas de trazabilidad y transparencia en la información por la aplicación subjetiva de su experiencia. Teniendo en cuenta lo anterior y además que el dominio SIG es conocido y que se pueden identificar sus tipos de datos y operaciones típicas, en este artículo se propuso un método de interoperabilidad que generaliza el estudio del dominio SIG. De esta manera, ante cada nuevo proyecto de interoperabilidad entre SIG, este método servirá como punto de partida en la fase de análisis. Adicionalmente, se incorporó el diseño

de patrones de análisis para los tres tipos de datos SIG identificados: *raster*, *vector* y series de datos y para once operaciones típicas que se llevan a cabo entre SIG. Estos patrones de análisis se diseñaron en esquemas preconceptuales, lo que permite obtener modelos conceptuales UML para las fases siguientes del desarrollo (diagramas de clases, de comunicación, máquinas de estado, de secuencias y casos de uso), además de otros artefactos comúnmente empleados en el desarrollo de aplicaciones de software (el diagrama de objetivos y las interfaces gráficas de usuario). Así se puede controlar la trazabilidad entre los diferentes elementos que arroja el método y se mejora la transparencia, pues el método actúa como una lista de chequeo que el analista debe seguir.

Las operaciones identificadas en el método son independientes del dominio de aplicación del SIG y del mecanismo de interoperabilidad que se seleccione para la implementación. Esta abstracción es la que permite que el método se pueda generalizar. Sin embargo, también sirven como punto de partida para operaciones de dominios específicos (p. ej., la obtención de una cuenca hidrográfica puede realizarse obteniendo el contorno de la malla de puntos que arroja el algoritmo de construcción de cuencas).

Como trabajo futuro que puede derivarse de este trabajo se propone:



- Incorporar en el proceso de desarrollo de los mecanismos para la interoperabilidad entre SIG taxonomías u ontologías que permitan llevar a cabo comparaciones entre nombres de capas y atributos automáticamente, sin la intervención del usuario. De esta manera se podría integrar el nivel semántico de la interoperabilidad con el sintáctico y funcional.
- Hacer un estudio detallado de la incorporación de las series de datos dentro de los SIG, identificar los diferentes tipos de series de datos que se pueden presentar, las series de tiempo y los perfiles de medición, así como su representación más conveniente en los SIG. También se recomienda identificar operaciones diferentes a las que se presentaron en este artículo y que involucren series de datos.
- Se recomienda incorporar los diferentes esquemas conceptuales UML que se pueden obtener a partir de los patrones de análisis diseñados mediante esquemas preconceptuales en un método de desarrollo de software intermedio SIG.
- Por fin, se recomienda, a partir de los patrones de análisis diseñados, incorporar un método de desarrollo de software que permita asistir paso a paso el desarrollo de un nuevo Sistema de Información Geográfica.

## AGRADECIMIENTOS

Este trabajo se financió con fondos del proyecto de investigación “Estudio de la problemática ambiental de tres embalses de EPM”, ejecutado en la Escuela de Geociencias y Medio Ambiente de la Universidad Nacional de Colombia, Sede Medellín, conjuntamente con la Universidad de Antioquia y las Empresas Públicas de Medellín.

## REFERENCIAS

Aburizaiza, Ahmad O. and Ames, Daniel P. (2009). *GIS-enabled desktop software development paradigms (sic)*. Proceedings of the 2009 International Conference on Advanced Geographic Information Systems & Web

Services (GEOWS '09), Cancun, Mexico (1-7 February), pp.75-79.

Al-Hader, Mahmoud; Rodzi, Ahmad; Sharif, Abdul Rashid and Ahmad, Noordin (2009). *SOA of smart city geospatial management*. Proceedings of the 3rd UKSim European Symposium on Computer Modeling and Simulation, Athens, Greece (25-27 November), pp. 6-10.

Bolstad, Paul V. *GIS fundamentals: A first text on geographic information systems*. 2<sup>nd</sup> ed. White Bear Lake, MN: Eider Press, 2005, 543 p.

Budiarto, Rahmat; Pradeep, Isawasan and Aziz, Maulana Abdul (2009). *Transformation of spatial data format for interoperability between GIS applications*. Proceedings of the Sixth International Conference on Computer Graphics, Imaging and Visualization, Tiajin, China (11-14 August), pp. 536-539.

Chen, Nengcheng; Di, Liping; Yu, Genong and Gong, Jianya (2010). “Geo-processing workflow driven wildfire hot pixel detection under sensor web environment”. *Computers & Geosciences*, vol. 36, No. 3 (March), pp. 362-372.

De Smith, Michael J.; Goodchild, Michael F. and Longley, Paul A. *Geospatial analysis: A comprehensive guide to principles, techniques and software tools*. 2<sup>nd</sup> ed. Leicester, UK: Troubador, 2007. 394 p. Available in: <http://www.spatialanalysisonline.com/>.

DeMers, Michael N. *GIS modeling in raster*. New Mexico: New Mexico State University, 2002. 203 p.

Du, Ying-jun; Yu, Chong-chong and Liu, Jie (2009). *A study of GIS development based on KML and Google Earth*. Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC, Seoul, Korea (25-27, August), pp. 1581-1585.

Fan, Hong; Feng, Hao and Zhu, Xinyan. *Integration of grid and OGC compliant services to implement the sharing and interoperability of multi-source and heterogeneous geospatial data*. Proceedings of the 17th International Conference on Geoinformatics, Fairfax, VA (12-14 August), pp. 1-6.

Fowler, Martin. *Analysis patterns: Reusable object models*. Menlo Park, CA: Addison Wesley Longman, 1997.

Hamza, Haitham and Fayad, Mohamed (2006). *Stable analysis patterns*. Proceedings of the 2006 IEEE International Conference on Computer Systems and Applications, pp. 545-548.

Hu, Chengfang; Di, Liping; Yang, Wenli; Wei, Yaxin; Bai, Yuqi; Lynnes, Cristopher; Enloe, Yonsook; Domenico, Ben and Rutledge, Glenn (2008). *Interoperability*

- middleware between geoscience and geospatial catalog protocols*. Proceedings of the 2008 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2008), Boston, MA (7-11 July), pp. 89-92.
- Iochpe, Cirano and Lisboa Filho, Jugurta (1998). *BCH/GIS: A basic class hierarchy to support GIS conceptual design*. Proceedings of the International Conference on Modeling Geographical and Environmental Systems with Geographical Information Systems (GIS), pp. 8-14.
- Lisboa Filho, Jugurta and Iochpe, Cirano (1999). *Specifying analysis patterns for geographic databases on the basis of a conceptual framework*. Proceedings of the ACM Symposium on Advances in Geographic Information Systems, pp. 7-13.
- Lisboa Filho, Jugurta; Iochpe, Cirano and Beard, Kate (1998). *Applying analysis patterns in the GIS domain*. Proceedings of the 10th Spatial Information Research Centre Colloquium, Dunedin, New Zealand (16-19 November), pp. 181-188.
- Lisboa Filho, Jugurta; Iochpe, Cirano and Borges, Karla (2002). "Analysis patterns for GIS data schema reuse on urban management applications". *CLEI Electronic Journal*, vol. 5, No. 2, pp.1-15.
- McDonald, Kenneth R.; Enloe, Yonsook; Di, Liping and Holway, Daniel (2006). *A gateway to support interoperability of OPeNDAP and OGC protocols*. Proceedings of the 2006 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2006), Denver, CO (31 July - 4 August), pp. 301-304.
- Rao, Ananth; Percivall, George and Enloe, Yonsook (2000). *Overview of the OGC catalog interface specification*. Proceedings of the IEEE 2000 International Geoscience and Remote Sensing Symposium (IGARSS 2000), vol. 3, pp.1211-1213.
- Rew, Russ and Davis, Glenn (1990). "NetCDF: An interface for scientific data access". *IEEE Computer Graphics and Applications*, vol. 10, No. 4 (July), pp. 76-82.
- Rozenstein, Offer and Karnieli, Arnon (2011). "Comparison of methods for land-use classification incorporating remote sensing and GIS inputs". *Applied Geography*, vol. 31, No. 2 (April), pp. 533-544.
- Sripada, Lakshmi N.; Lu, Chang-Tien and Wu, Weili (2004). *Evaluating GML support for spatial databases*. Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC 2004), Hong Kong, China (28-30 September), vol. 2, pp. 74-77.
- Sunyaev, Ali; Hansen, Mathias and Krcmar, Helmut (2010). *Method engineering: a formal description*. Papadopoulos, G. A.; Wojtkowski, W.; Wrycza, S. and Zupancic, J. (eds.), *Information systems development towards a service provision society*. Berlin: Springer, 2010, pp. 645-654.
- Xu, Chao; Li, Sha and Mi, Pu-chun (2010). *OPeNDAP based physical oceanographic data service system of South China*. Proceedings of the 2010 International Conference on Innovative Computing and Communication and 2010 Asia-Pacific Conference on Information Technology and Ocean Engineering (CICC-ITOE 2010), Macao, China (30-31 January), pp. 54-57.
- Yan, Yunpeng; Li, Jiancun and He, Zhengmin (2006). *Research on ontology based semantic integration model in spatial data sharing*. Proceedings of the 2006 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2006), Denver, CO (31 July - 4 August), pp. 2872-2875.
- Zapata, Carlos Mario; Gelbukh, Alexander and Arango Isaza, Fernando (2006) "Pre-conceptual schema: A conceptual-graph-like knowledge representation for requirements elicitation". *Lecture Notes in Computer Science*, vol. 4293, pp. 17-27.
- Zapata, Carlos Mario; Tamayo, Paula y Arango, Fernando (2007). "Conversión de esquemas preconceptuales a diagramas de casos de uso en AToM3". *Dyna*, vol. 74, No.153 (septiembre-diciembre), pp. 237-251.
- Zhang, Shuliang; Gan, Jiayan; Miao, Lizhi; Lv, Guonian and Huang, Jiazhu (2007). *Study on GML spatial interoperability based on web service*. Proceedings of the 31st Annual International Conference on Computer Software and Applications, 2007. COMPSAC 2007. vol. 1. pp. 649-656.