Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ciencias de la Computación e Informática

Informe de Proyecto de Graduación para optar por el grado académico de
Licenciatura en Computación e Informática

GeoCR: una aplicación web de análisis dinámico para el soporte de decisiones basadas en datos convencionales y espaciales

Diana Bogantes González - A60863
Leonardo Pandolfi González - A64277

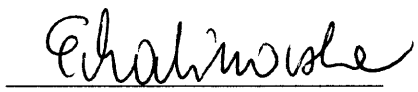
Ciudad Universitaria Rodrigo Facio Brenes

San José, Costa Rica

Febrero, 2014

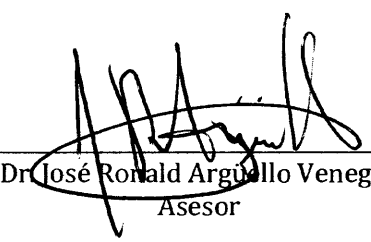
Este proyecto de graduación ha sido aceptado por el Tribunal Examinador como requisito parcial para optar al grado académico de Licenciatura en Computación e Informática.

Miembros del Tribunal Examinador:

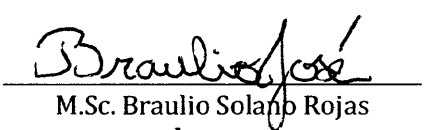


Dra. Elzbieta Malinowski Gadja
Directora del proyecto

Dr. Gilbert Brenes Camacho
Asesor



Dr. José Ronald Arguello Venegas
Asesor



M.Sc. Braulio Solano Rojas
Lector

Dr. Arturo Camacho Lozano
Representante del Director de la Escuela de
Ciencias de la Computación e Informática

DEDICATORIAS

A Dios por brindarme sabiduría y fortaleza en cada momento de la vida.

A mis padres y hermana por sus consejos y apoyo incondicional todos estos años. Son mi guía y mi soporte y les estoy eternamente agradecida.

A la memoria de dos personas muy especiales que a pesar de que no tuve la dicha de compartir este logro con ellos, están siempre en mi corazón y sé que desde el cielo están disfrutando conmigo.

Diana

•••

En un mundo donde los halagos baratos son frecuentes y las dedicatorias siguen un formato preestablecido, dedico esto a quienes me impulsan a apuntar siempre a lo mejor por medio de sus críticas honestas; a mis padres Giselle González Arce y Leonardo Pandolfi Lizano. Sin ellos y su orientación al perfeccionismo, este pequeño espacio estaría ocupado por palabras demasiado similares a las de otros mil trabajos finales de graduación. Infinitas gracias.

Como fuente de sabiduría y consejos que alientan la voluntad para perseverar, lo dedico también a mis abuelos Luz Marina Lizano Retana y Gonzalo González Coto. Gracias por permitirme ser oído de sus anécdotas y receptor de sus conocimientos. De una forma extraña, todo eso está contenido en los esfuerzos que produjeron este proyecto. Aunque uno de ustedes nunca llegará a leer esto, confío en que otros sabrán valorar este ínfimo reconocimiento.

Finalmente, lo dedico a mis hermanos. Esto es también fruto de las risas, discusiones y peleas constantes con ustedes. Como dos personas que sin duda me superan, les agradezco que me hayan entendido como el tipo competitivo e idealista, que trata de compensar sus deficiencias con un corazón peleador. Espero que me permitan seguir incluyendo una cuota de fastidio en sus vidas por mucho tiempo más, siempre haciendo el papel de Gattuso a la par de Pirlo y Kaká. A mis amigos de toda la vida, Chelo y Mau.

Leonardo

AGRADECIMIENTOS

A la Dra. Elzbieta Malinowski Gajda por su guía y abierta disposición para la culminación exitosa del presente trabajo; por retarnos constantemente a subir nuestro nivel de exigencia y por creer en nosotros.

Al Dr. Gilbert Brenes Camacho por su incondicional apoyo y confianza; por impulsar el proyecto y estar siempre anuente a brindar su valiosa ayuda.

Al Bach. Jairo Sosa Mesén por sugerir la idea de llevar a cabo este proyecto y convertirlo en el trabajo final de graduación; por su colaboración atenta a lo largo de todo este proceso.

Al M.Sc. Daniel Antich Montero por la atención prestada; por el tiempo empleado en la elaboración de sugerencias que mejoraron el producto final del trabajo.

Al Bach. Andrés Ramírez Gutiérrez por su contribución con la limpieza de los datos espaciales y por su ayuda en la generación de ciertas imágenes empleadas en este documento.

Y, finalmente, a todas las personas que de una u otra forma influyeron positivamente en el desarrollo del proyecto; al personal del Centro Centroamericano de Población, a nuestros familiares y amigos, por su apoyo, tanto explícito como tácito.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE EXTRACTOS DE CÓDIGO

ÍNDICE DE ABREVIATURAS

La siguiente lista muestra las abreviaturas y acrónimos usados en este documento:

BI:	Inteligencia de Negocios (*Business Intelligence*)
CCP:	Centro Centroamericano de Población
CCSS:	Caja Costarricense de Seguro Social
DBMS:	Sistema de Administración de Base de Datos (*Database Management System*)
DSS:	Sistema de Soporte de Decisiones (*Decision Support System*)
DW:	Almacén de Datos (*Data Warehouse*)
ETL:	Extracción, Transformación y Carga (*Extraction, Transformation, and Load*)
INEC:	Instituto Nacional de Estadística y Censos
MDX:	Expresiones Multidimensionales (*MultiDimensional eXpressions*)
OLAP:	Procesamiento Analítico en Línea (*On-Line Analytical Processing*)
SDW:	Almacén de Datos Espacial (*Spatial Data Warehouse*)
SOLAP:	Procesamiento Analítico en Línea de datos espaciales (*Spatial On-Line Analytical Processing*)
SRID:	Identificador de Referencia Espacial (*Spatial Reference Identifier*)
WGS84:	Sistema Geodésico Mundial 84 (*World Geodetic System 84*)

RESUMEN

Cita bibliográfica

Bogantes G., Diana & Pandolfi G., Leonardo. (2014). *GeoCR: una aplicación web de análisis dinámico para el soporte de decisiones basadas en datos convencionales y espaciales*. Informe de Proyecto de Graduación para optar por el grado de Licenciatura en Ciencias de la Computación e Informática. Ciudad Universitaria Rodrigo Facio Brenes, Universidad de Costa Rica.

Directora del proyecto

Dra. Elzbieta Malinowski Gajda

Palabras clave

Almacenes de datos espaciales, SOLAP, cubo, medida, dimensión, cubos virtuales, medidas calculadas

Resumen

El informe del Trabajo Final de Graduación, denominado "GeoCR: una aplicación web de análisis dinámico para el soporte de decisiones basadas en datos convencionales y espaciales", profundiza en la implementación de un sistema que permite realizar consultas *ad-hoc* sobre nacimientos, defunciones e incidencia de cáncer en Costa Rica. Para el desarrollo de la aplicación, se propuso la creación de un sistema de Inteligencia de Negocios

(BI – *Business Intelligence*) que, utilizando almacenes de datos espaciales y herramientas SOLAP, permitiera el despliegue de resultados en tablas, mapas y gráficos en forma simultánea y sincronizada.

Los almacenes de datos son elementos básicos para las soluciones BI. Ellos permiten el almacenamiento de datos históricos relacionados con los enfoques de análisis que se consideren relevantes, así como la integración de datos de distintos sistemas operacionales y fuentes externas. En el informe del proyecto se detallan las diferentes etapas del desarrollo de un almacén de datos espacial, el cual incorpora un elemento anexo para permitir la representación de objetos espaciales, tales como geometrías. De igual forma, se detallan las transformaciones que se debieron aplicar a los datos originarios de las diferentes fuentes externas antes de su carga en el almacén.

Los servidores SOLAP son los encargados del procesamiento de consultas para acceder a los datos cargados en el almacén. En el informe se detalla la creación del esquema XML usado por GeoMondrian, motor SOLAP seleccionado. Asimismo, se explican los pasos para implementar estructuras complejas como dimensiones compartidas, medidas calculadas y cubos virtuales en GeoMondrian, necesarias para cumplir con los requerimientos del proyecto.

Debido al requerimiento del despliegue de resultados en mapas, se seleccionó la herramienta cliente SOLAP denominada GeoOLAP, la cual tiene la capacidad de desplegar los resultados en tablas, gráficos y mapas de forma simultánea y sincronizada. En el informe se detalla la forma en que esta fue modificada para ajustarla a las necesidades del CCP. De igual forma, se

explican diferentes escenarios de análisis empleados para validar el adecuado funcionamiento del sistema, así como la veracidad de los datos resultantes.

Por último, se muestran los problemas encontrados durante la implementación de cada una de las etapas del proyecto. Asimismo, se proponen diferentes posibilidades de mejora, particularmente en GeoOLAP, para extender las funcionalidades actuales y ampliar el análisis que se pueden realizar con la herramienta. La aplicación implementada se encuentra disponible en el sitio web del CCP (http://ccp.ucr.ac.cr/tasas_demograficas/tasas.html).

CAPÍTULO I: INTRODUCCIÓN

1. JUSTIFICACIÓN

A través de los años, distintas instituciones nacionales han recopilado datos acerca de nacimientos, defunciones e incidencia de cáncer en Costa Rica. El Instituto Nacional de Estadística y Censos (INEC) y el Ministerio de Salud son dos de esos organismos, los cuales también manejan una segmentación del territorio nacional propia, confeccionada de acuerdo con sus focos de interés para el análisis de los datos colectados. Por ejemplo, la Caja Costarricense de Seguro Social (CCSS) cuenta con una división geográfica por regiones y áreas de salud, mientras que el INEC maneja regiones y subregiones de planificación. Además, estas instituciones recolectan otros tipos de datos —como los relacionados con el género y los grupos de edad— que, en conjunto, proveen potencial para un análisis variado. Gracias a convenios con las entidades mencionadas, el Centro Centroamericano de Población (CCP) pone a disposición de los usuarios la posibilidad de consultar estos datos a través de su plataforma web; asimismo, estos acuerdos le permiten recibir nuevos datos periódicamente, por lo que mantiene sus bases de datos actualizadas.

Actualmente, el sitio web del CCP ofrece el Sistema de Consulta a Bases de Datos Estadísticas[1], que cuenta con quince bases de datos sobre censos, nacimientos, defunciones y el registro de tumores. No obstante, cada una de ellas debe ser consultada por separado para

[1] Centro Centroamericano de Población, "Sistema de Consulta a Bases de Datos Estadísticas". [En línea]. Disponible: http://censos.ccp.ucr.ac.cr/ [Último acceso: 2 de febrero, 2014]

obtener la información deseada. Por ejemplo, la Figura 1 muestra el menú de selección de base de datos, que ilustra el impedimento para elegir más de una de ellas en la misma consulta. Por su parte, las Figuras 2 y 3 ejemplifican las pantallas para la formulación de consultas y la creación de expresiones o filtros, correspondientemente. El proceso de selección —tanto de operadores como de variables para las consultas— requiere un conocimiento previo del sistema. En caso de que el usuario no cuente con él, es necesario que se dirija a la sección de ayuda para aprender cómo debe hacer la consulta y qué posibilidades le brinda la interfaz.

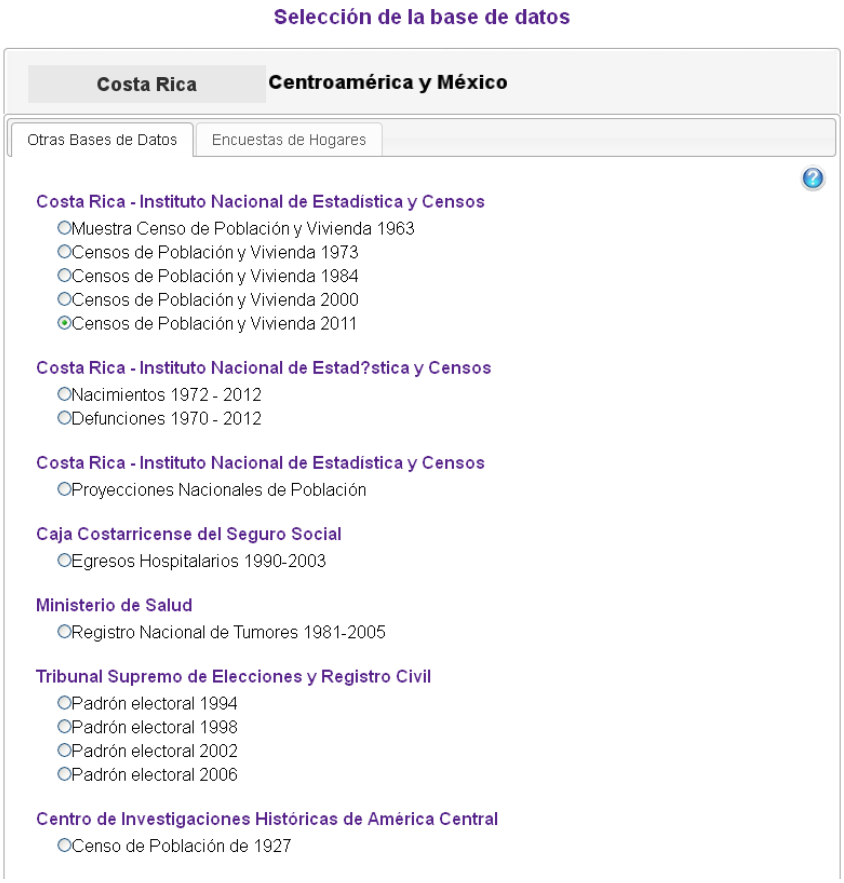


Figura 1. Selección de bases de datos para consultar [CEN12]

Figura 2. Formulación de consultas [CEN12]

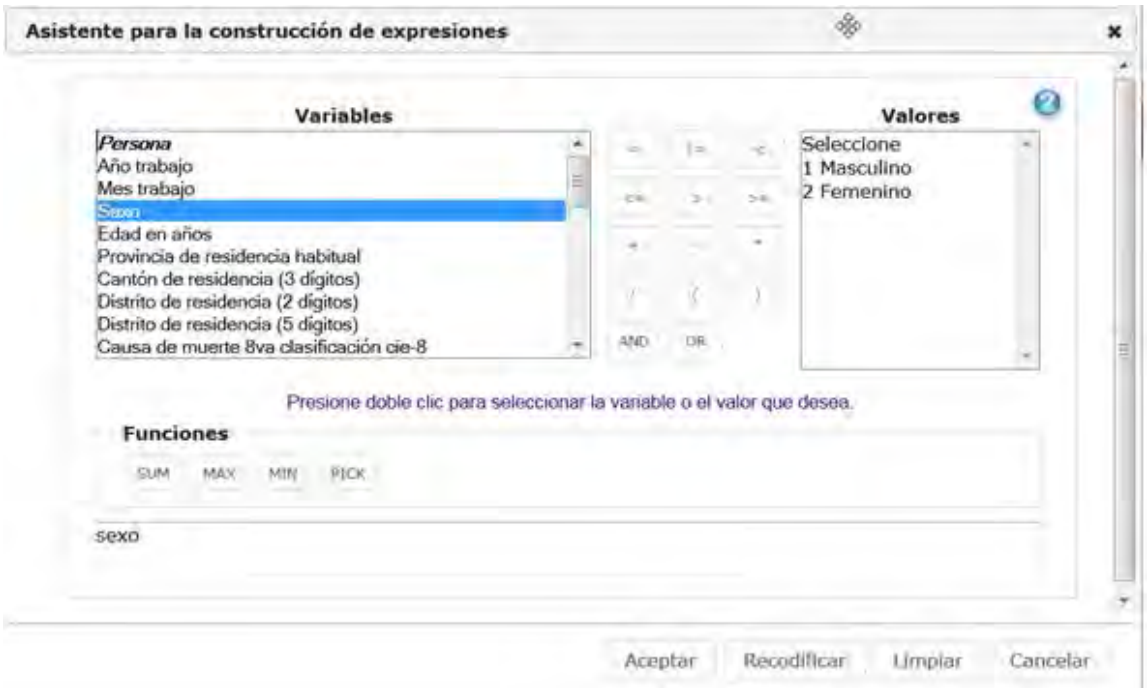
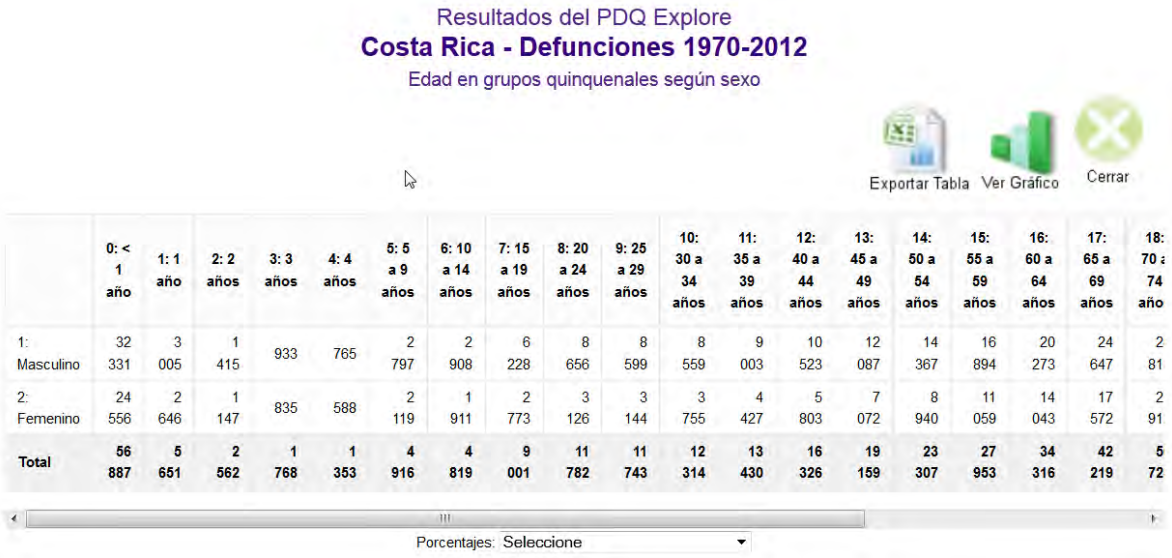


Figura 3. Formulación de expresiones o filtros para la consulta [CEN12]

El Sistema de Consulta a Bases de Datos Estadísticas del CCP permite ver los resultados generados por las consultas mediante tablas y gráficos. Por ejemplo, si se ejecuta la consulta con las variables seleccionadas en la Figura 2, se obtienen la tabla y el gráfico mostrados en las Figuras 4 y 5. Estos resultados son estáticos, por lo que si se desea agregar alguna otra variable u obtener más detalle sobre algún parámetro, se debe crear una consulta nueva.



	0: < 1 año	1: 1 año	2: 2 años	3: 3 años	4: 4 años	5: 5 a 9 años	6: 10 a 14 años	7: 15 a 19 años	8: 20 a 24 años	9: 25 a 29 años	10: 30 a 34 años	11: 35 a 39 años	12: 40 a 44 años	13: 45 a 49 años	14: 50 a 54 años	15: 55 a 59 años	16: 60 a 64 años	17: 65 a 69 años	18: 70 a
1: Masculino	32 331	3 005	1 415	933	765	2 797	2 908	6 228	8 656	8 599	8 559	9 003	10 523	12 087	14 367	16 894	20 273	24 647	2 81
2: Femenino	24 556	2 646	1 147	835	588	2 119	1 911	2 773	3 126	3 144	3 755	4 427	5 803	7 072	8 940	11 059	14 043	17 572	2 91
Total	**56 887**	**5 651**	**2 562**	**1 768**	**1 353**	**4 916**	**4 819**	**9 001**	**11 782**	**11 743**	**12 314**	**13 430**	**16 326**	**19 159**	**23 307**	**27 953**	**34 316**	**42 219**	**5 72**

Figura 4. Resultados en tabla [CEN12]

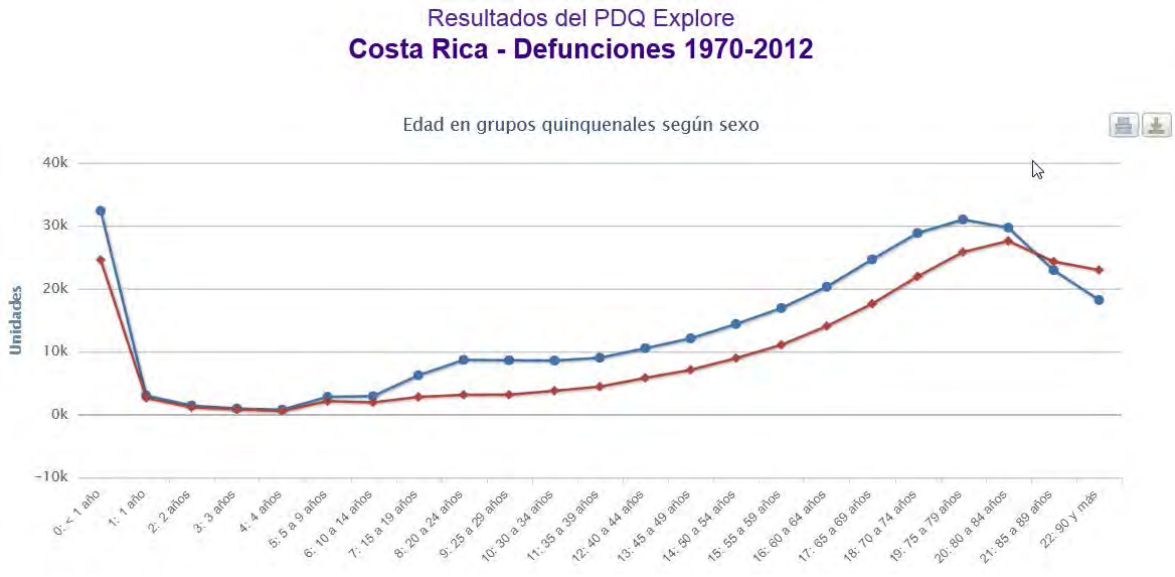


Figura 5. Resultados en gráfico [CEN12]

El CCP también cuenta con el sistema de consulta InfoCensos[2], que permite obtener el despliegue de estadísticas sobre el mapa de Costa Rica para diecinueve temas distintos, tal y como se muestra en la Figura 6. Adicionalmente, al pulsar algún cantón, esta herramienta despliega gráficos asociados en formato de imagen, como se observa en la Figura 7. Sin embargo, los resultados de InfoCensos son estáticos y limitados porque (1) sólo permite seleccionar una única variable (a la que el sistema llama "tema"), (2) los resultados del mapa son exclusivos para el año 2000 y (3) la única división territorial que ofrece es la cantonal.

[2] Centro Centroamericano de Población, "InfoCensos". [En línea]. Disponible: http://infocensos.ccpucr.ucr.ac.cr/ [Último acceso: 2 de febrero, 2014]

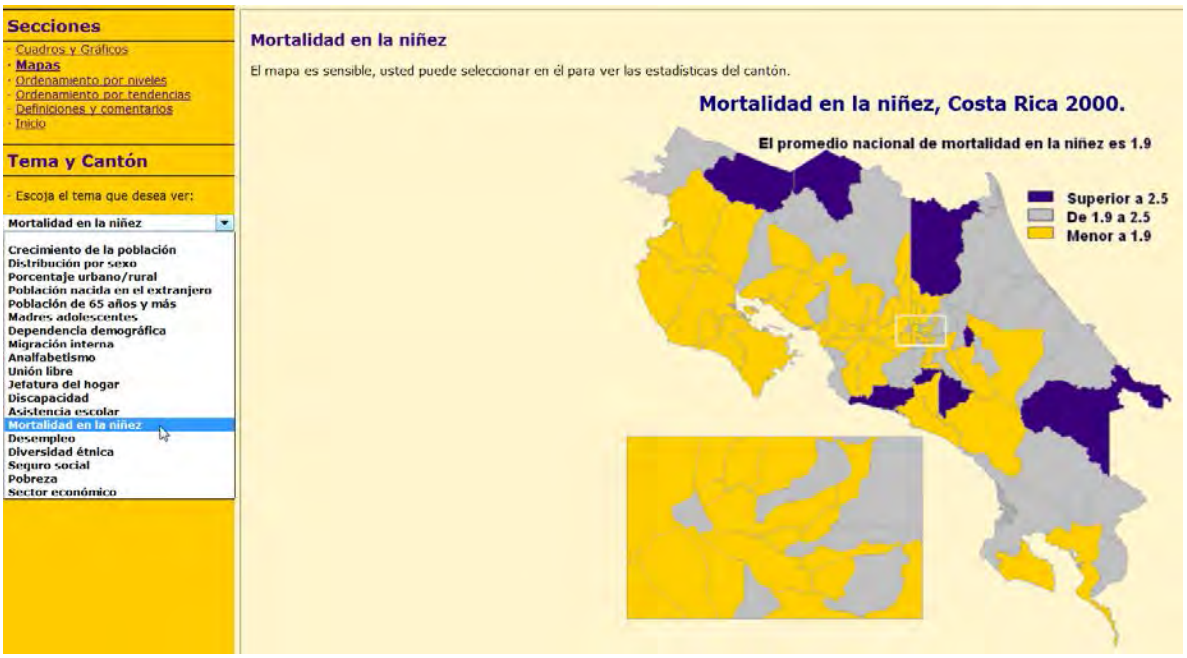


Figura 6. Ejemplo de mapas actuales para defunciones en la niñez [CEN13]

Figura 7. Gráfico mostrado al pulsar el cantón de Heredia en el mapa [CEN13]

Como se ha expuesto, las opciones presentes en la plataforma web del CCP permiten consultar los datos existentes sobre distintas temáticas. No obstante, la falta de integración entre las bases de datos, la complejidad en la formulación de las consultas, la rigidez de los resultados y la incapacidad para mostrar las formas de despliegue de los datos simultáneamente (tablas, gráficos y mapas), hicieron que el CCP planteara la necesidad de tener un repositorio menos particionado y con aquellos datos que, según la consideración de sus investigadores, fueran los necesarios para realizar los análisis deseados. Como respuesta a estas carencias, se planeó el desarrollo de un sistema más dinámico e intuitivo que les permitiera a los usuarios formular consultas *ad-hoc* y poder elegir entre los datos disponibles para combinarlos libremente. Así, el sistema permitiría observar y manipular los resultados en tablas, mapas y gráficos en forma simultánea y sincronizada.

La implementación de la propuesta del CCP dio origen al proyecto GeoCR, el cual utiliza un subconjunto de los datos de la CCSS, el INEC y el CCP sobre incidencia de cáncer, defunciones, defunciones infantiles y nacimientos, con el objetivo de permitir su análisis a lo largo de diferentes perspectivas, como grupos de edad, género, tiempo y distribución geográfica. Los datos, almacenados en un Almacén de Datos Espacial (SDW - *Spatial Data Warehouse*), son utilizados por herramientas de Procesamiento Analítico Espacial en Línea (SOLAP - *Spatial On-Line Analytical Processing*) para el procesamiento y despliegue de resultados. El alcance del sistema y la selección de la muestra de datos están determinados por los requerimientos del CCP y las especificaciones del personal de esta institución.

El uso de tecnologías no tradicionales en el proyecto —como lo son SOLAP y SDW— permite que los usuarios puedan, por ejemplo, consultar la incidencia de un tipo de cáncer por provincia en el último año y, posteriormente, añadirle el componente de género, para determinar la incidencia de ese tipo de cáncer en hombres y mujeres por aparte. A través de estas herramientas, los datos pueden ser visualizados en tablas, gráficos y mapas, facilitando su interpretación y uso.

Los resultados obtenidos en GeoCR pueden utilizarse para realizar análisis exploratorios con el fin de detectar patrones, descubrir relaciones entre variables y ayudar en la toma de decisiones de interés local o nacional; esto sin la necesidad de conocimiento especializado en estadística o informática. Consecuentemente, el sistema sirve como base para otras instituciones o individuos que, como el CCP, deseen utilizar y analizar información demográfica, epidemiológica y territorial.

2. OBJETIVOS

OBJETIVO GENERAL

- Desarrollar una aplicación web basada en los conceptos del modelo multidimensional espacial para el análisis dinámico de una muestra de datos demográficos de Costa Rica.

OBJETIVOS ESPECÍFICOS

- Seleccionar, limpiar e integrar una muestra de datos demográficos para el desarrollo del proyecto.
- Diseñar e implementar un almacén de datos espacial para la muestra de datos.
- Crear cubos SOLAP que utilicen datos demográficos y espaciales del almacén de datos.
- Incorporar los cubos SOLAP al sitio web del CCP.
- Crear diferentes escenarios de análisis utilizando los cubos espaciales creados.

3. DELIMITACIÓN DEL PROBLEMA

El CCP cuenta con una variedad de registros distribuidos en distintas bases de datos. Por ejemplo, se tiene información sobre registros de tumores, defunciones, padrones electorales, población y vivienda. Sin embargo, no toda esta información es relevante para los objetivos de este proyecto. Por lo tanto, de los datos disponibles se tomará el subconjunto necesario para permitir el análisis requerido por los usuarios, relacionado con cáncer, defunciones, defunciones infantiles y nacimientos.

El sistema tendrá como base herramientas de *software* libre, una medida determinada así por el CCP. Esto concuerda con el impulso que le ha dado el gobierno costarricense a la utilización de *software* libre, mismo que se ha hecho manifiesto al darle prioridad sobre el *software* propietario en las instituciones públicas [ASA12]. Un aspecto por considerar es que, a pesar de que las herramientas de acceso libre para el análisis de datos poseen distintas funcionalidades, estas se ven reducidas cuando se incorpora el elemento espacial. Lo anterior, sumado a la limitada variedad de herramientas SOLAP libres disponibles, hace que las consultas sean delimitadas por las funcionalidades que las soluciones ofrecen.

CAPÍTULO II: MARCO TEÓRICO

1. SISTEMAS DE SOPORTE DE DECISIONES E INTELIGENCIA DE NEGOCIOS

Con el pasar de los años, diversas compañías con largas trayectorias en el mercado descubrieron que los datos históricos que tenían almacenados podían servirles para determinar, por ejemplo, mejoras en las cadenas de distribución, posibles socios comerciales y evaluar programas de financiamiento y patrocinio. Muestras de este tipo de beneficios pusieron en evidencia la falta de un proceso automatizado que permitiera el aprovechamiento de los datos. Así, con el objetivo de cumplir sus metas de desarrollo, las organizaciones buscaron emplear los datos recopilados a través de los años en análisis para la toma de decisiones, el descubrimiento de patrones y en la mejora de la productividad en general [MAL08].

El concepto de **Sistema de Soporte de Decisiones** (DSS - *Decision Support Systems*) surgió en la década de los 70 para describir aplicaciones computacionales que analizan datos del negocio y los presentan a los usuarios para ayudarles en la toma de decisiones [KOP09]. El objetivo de los DSS fue suplir la necesidad de tener sistemas computacionales que, tomando como base la definición de un problema y un conjunto de datos de entrada, facilitaran la formulación de soluciones.

Posteriormente, surgió el término **Inteligencia de Negocios** (BI – *Business Intelligence*), definido como el proceso de transformar datos en información, información en inteligencia, inteligencia en conocimiento y conocimiento en sabiduría de negocio [KOP09]. Para poder lograrlo, combina aplicaciones y tecnologías relacionadas con la recopilación, el almacenamiento y el análisis de los datos, así como una interfaz de acceso al usuario para asistir en la toma de decisiones estratégicas y operacionales. Como consecuencia de su carácter interdisciplinario, los sistemas de BI comenzaron a remplazar a los DSS para aprovechar los recursos internos y externos de información de una organización, con el fin de mejorar las decisiones de negocio [KIM11] mediante el uso de conceptos como almacenes de datos, minería de datos y procesamiento analítico en línea [KOP09], tal como se muestra en la Figura 8.

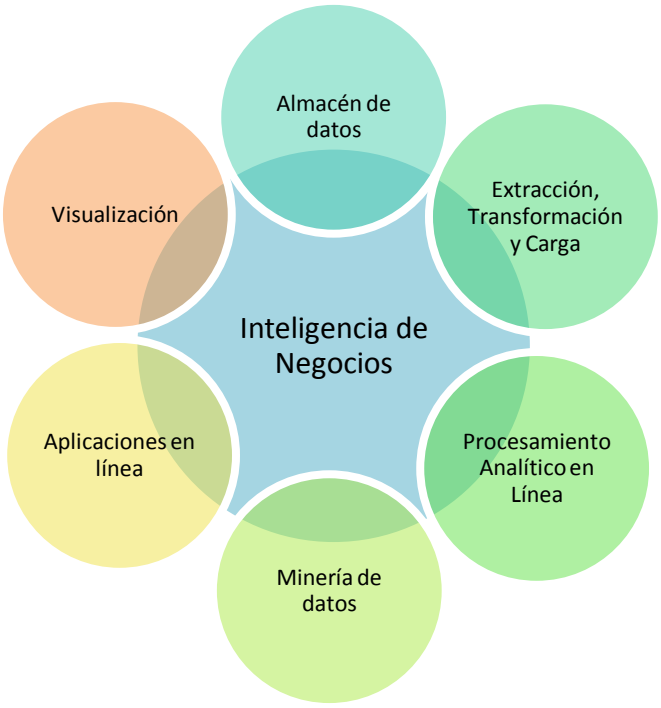


Figura 8. Componentes de BI

Los almacenes de datos en conjunto con herramientas de **Procesamiento Analítico en Línea** (OLAP – *On Line Analytical Processing*), constituyen la base de una solución de BI al servir, respectivamente, como fuentes de almacenamiento para los datos históricos y motores de consulta especializados para el análisis de los mismos. Otras tecnologías, como las mostradas en la Figura 8, también pueden ser integradas dentro de un sistema de BI. En las secciones subsiguientes de este capítulo se profundizará en estos conceptos, junto con los procesos de **Extracción, Transformación y Carga** (ETL - *Extraction, Transformation and Load*) requeridos en la creación de un almacén de datos.

2. ALMACENES DE DATOS

Los problemas emergentes al almacenar y consultar datos históricos —propios de las soluciones de BI— no fueron fáciles de remediar usando las soluciones disponibles para bases de datos operacionales y transaccionales. Estas opciones generalmente tienen un mal desempeño al ejecutar consultas complejas que requieren la unión de varias tablas, pues poseen un diseño altamente normalizado. Además, no fueron concebidas para almacenar datos históricos por largos períodos de tiempo, lo cual impide que se pueda realizar un análisis histórico del comportamiento de una organización. Por consiguiente, si bien las bases de datos mencionadas son capaces de sobrellevar las operaciones diarias para permitir el acceso rápido a los datos y soportar transacciones de diferentes usuarios concurrentes, no satisfacen los requerimientos básicos para permitir el análisis de datos [MAL08].

Los **Almacenes de Datos** (DW - *Data Warehouse*) fueron propuestos como producto de las limitaciones anteriores. El concepto surgió en la década de los 90 y desde entonces se ha consolidado como un componente base de BI [INM96]. El término "almacén de datos" —propuesto por W.H. Inmon [INM96, INM05]— se define como una colección de datos utilizada a manera de apoyo para la toma de decisiones, que cuenta con las siguientes características [INM05, MAL08]:

- **Orientada al análisis.** Los almacenes se concentran en los focos de análisis requeridos por la organización; por ejemplo, en el caso del CCP, la incidencia de los tipos de cáncer y el comportamiento de las tasas de defunciones y natalidad en Costa Rica. Un mismo almacén puede incluir distintos enfoques propuestos dentro de la organización.
- **Integrada.** Reúne datos de distintos sistemas operacionales y fuentes externas. Para lograr consistencia en los datos, se necesita resolver las diferencias en la definición de datos y contenido.
- **No volátil.** No se permite la modificación ni la eliminación de datos, por lo que se pueden abarcar periodos más largos que en los sistemas operacionales.
- **Variante en el tiempo.** Es posible almacenar diferentes valores para el elemento de interés agregando la variable del tiempo. Por ejemplo, se puede almacenar el número total de nacimientos en distintos años y además se puede conocer el momento en que ocurrió un cambio en los valores.

Los datos en un almacén de datos son recolectados y acumulados a lo largo del tiempo, con el objetivo de estudiar su evolución, encontrar patrones y analizar correlaciones [MAL08]. Consecuentemente, los DW se diseñan e implementan para facilitar la realización de consultas complejas. Estos aspectos y otros adicionales —mostrados en la Tabla 1 [INM05, MAL08]— establecen la diferencia entre un almacén de datos y una base de datos operacional.

Tabla 1. Diferencias entre bases de datos operacionales y almacenes de datos [INM05, MAL08]

Aspecto	Bases de datos operacionales	Almacenes de datos
Tiempo de almacenamiento	60 a 90 días	5 a 10 años
Elemento de tiempo	Puede o no contenerlo	Lo contiene siempre
Tipo de usuario	Operadores y empleados de oficina	Gerentes y ejecutivos de alto nivel
Datos contenidos	Del presente y detallados	Históricos y agregados
Uso	Predecible y repetitivo	*Ad-hoc* y sin estructura
Frecuencia de acceso	Alta	Media a baja
Tipo de acceso	Lectura, actualización, borrado e inserción	Lectura e inserción
Número de registros por acceso	Pocos	Muchos

Tiempo de respuesta	Bajo	Puede ser alto
Nivel de concurrencia	Alto	Bajo
Redundancia de datos	Poca (tablas normalizadas)	Alta (tablas no normalizadas)
Modelado de datos	Entidad-Relación	Multidimensional

ARQUITECTURA DE UN ALMACÉN DE DATOS

Para poder cubrir todas las funcionalidades descritas en la Tabla 1 y como parte de la creación de un almacén de datos, se necesita definir una arquitectura que sirva de soporte para todo el proceso de integración y análisis de datos que comprende una solución de BI. La Figura 9 presenta la arquitectura típica de un almacén de datos [MAL08].

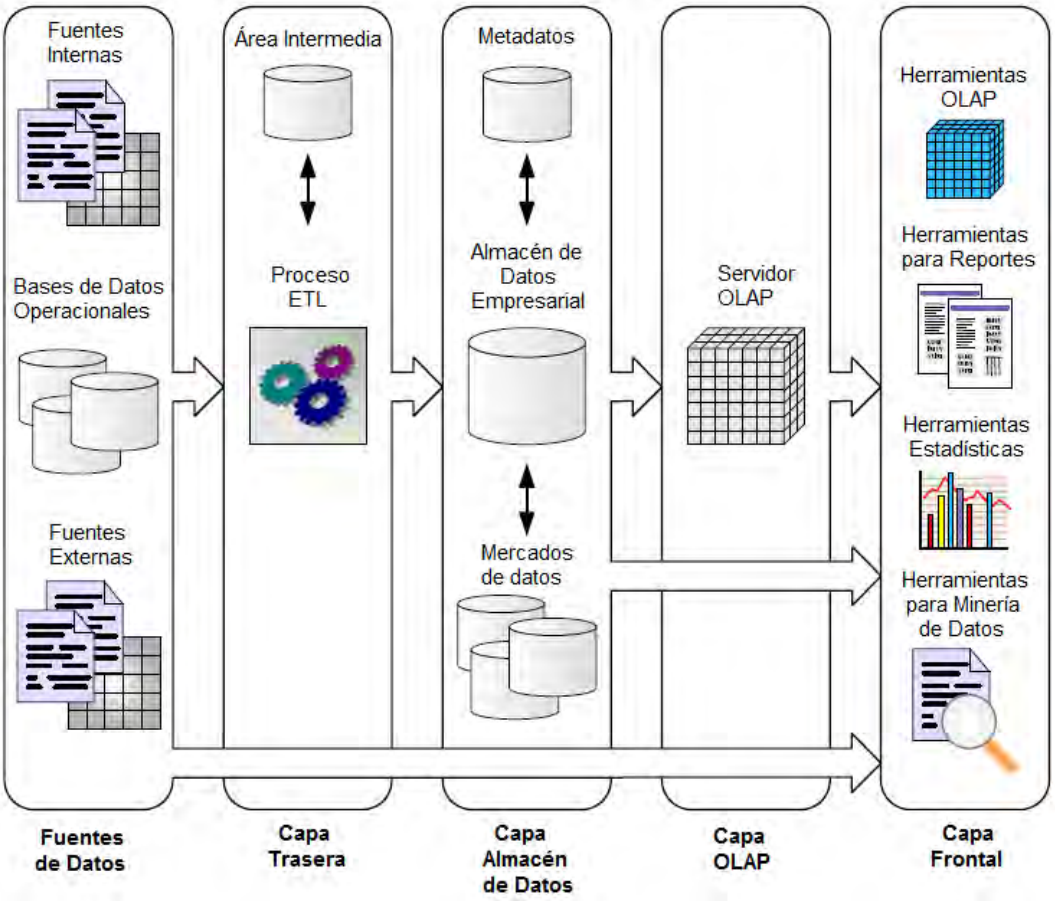


Figura 9. Arquitectura típica de un almacén de datos [MAL08]

Capa back-end (*Back-end Tier*)

La capa de *back-end* está compuesta por las herramientas responsables del proceso de ETL, que son utilizadas para alimentar al almacén con datos extraídos de archivos, bases de datos operacionales y otros orígenes. En algunos casos, al aplicar los procesos de transformación, se puede requerir un área intermedia de almacenamiento conocida como *staging area* [KIM11], que también forma parte de la capa *back-end*. Una vez que los datos han pasado por las transformaciones requeridas, son cargados en el almacén.

Capa Almacén de Datos (*Data Warehouse Tier*)

La capa de almacenamiento de datos está compuesta por el almacén de datos (DW), mercados de datos (*data marts*) y un repositorio de metadatos. El DW contiene toda la información recopilada y transformada, lista para ser analizada. Los mercados de datos son almacenes más pequeños y especializados que se enfocan en áreas específicas de la compañía o en un grupo de usuarios en particular. El repositorio de metadatos contiene información sobre el DW y su contenido, incluyendo el detalle sobre los datos disponibles, su ubicación y los procesos implementados.

Capa OLAP (*OLAP Tier*)

La capa de OLAP está formada por el servidor OLAP, el cual soporta datos y operaciones especiales para el análisis dinámico de los datos. Existen diferentes tipos de servidores OLAP que se diferencian por su forma de almacenamiento físico. Entre ellos están [MAL08]:

- **OLAP Relacional** (ROLAP - *Relational OLAP*). Almacena los datos en bases de datos relacionales. Utiliza extensiones SQL y diferentes métodos de acceso para implementar el modelo de datos multidimensional —explicado en la siguiente sección— y sus operaciones.
- **OLAP Multidimensional** (MOLAP - *Multidimensional OLAP*). Almacena directamente los datos multidimensionales en estructuras especiales —como arreglos— e implementa operaciones OLAP sobre esas estructuras. Provee menor capacidad de almacenamiento que ROLAP, pero es más eficiente al ejecutar consultas y realizar la agregación de datos.

- **OLAP Híbrido** (HOLAP - *Hybrid OLAP*). Combina ROLAP y MOLAP, proporcionando una amplia capacidad de almacenamiento en bases de datos relacionales, mientras que las funciones de agregación se manejan usando estructuras MOLAP.
- **OLAP Espacial** (SOLAP - *Spatial OLAP*). Expande las funcionalidades y características de ROLAP incorporando el almacenamiento y manipulación de datos espaciales como, por ejemplo, la geometría de una provincia.

Capa front-end (Front-end Tier)

La capa de *front-end* abarca lo relacionado con el análisis y la visualización de los datos. Incluye herramientas que permiten formular consultas complejas de forma intuitiva para el usuario, descubrir patrones y tendencias, manipular los resultados de las consultas en forma interactiva (como en el caso de las herramientas OLAP) y realizar predicciones con base en los datos actuales, mediante herramientas para la minería de datos.

DISEÑO DE UN ALMACÉN DE DATOS: MODELO MULTIDIMENSIONAL

Para implementar el almacén de datos, que forma parte de la arquitectura en capas mostrada en la Figura 9, se necesita primero diseñar un esquema acorde con los requerimientos de análisis particulares. La práctica común en el desarrollo de DW es utilizar el **modelo multidimensional**, porque este supera al modelo entidad-relación en términos de expresión de consultas complejas durante el análisis [MAL08].

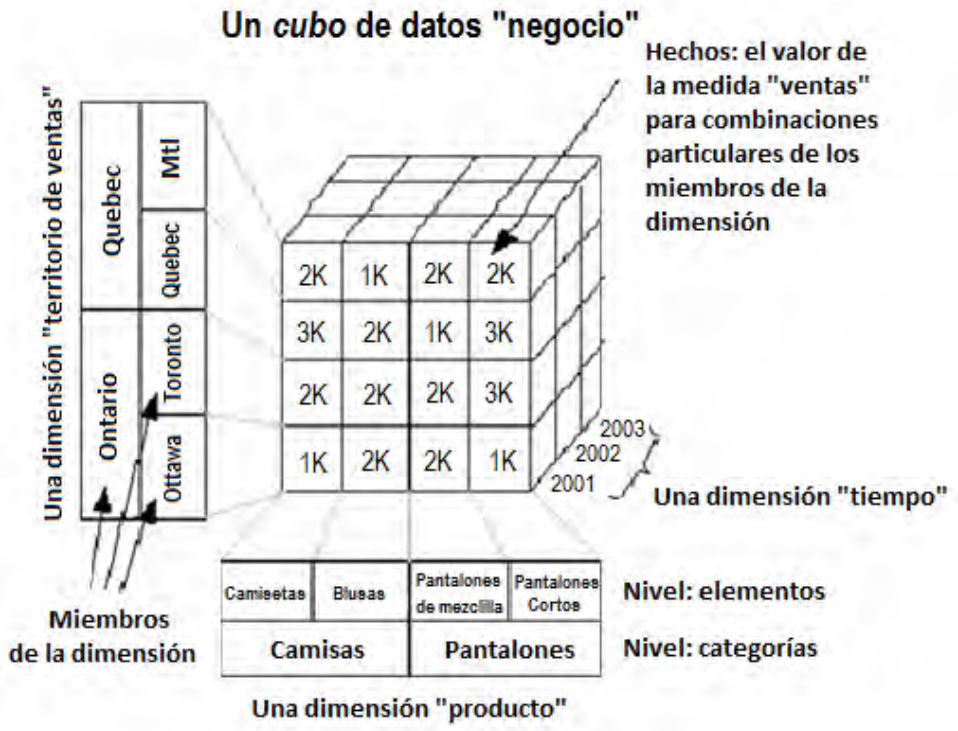


Figura 10. Cubo en un modelo multidimensional [RIV03]

El modelo multidimensional se basa en una visión abstracta del objeto de análisis llamada **cubo**, que está formado por medidas y dimensiones. Una **medida** corresponde a un hecho del negocio que se desea analizar. Por ejemplo, en la Figura 10 se muestra un cubo denominado Negocio que contiene la medida Ventas para representar la cantidad de unidades vendidas por producto en un determinado territorio durante un año. Por otra parte, una **dimensión** es un conjunto de atributos que sirven para identificar y categorizar medidas desde distintas perspectivas. Siguiendo el contexto de ventas de la Figura 10, tres posibles dimensiones serían Territorio de ventas, Producto y Tiempo. A las instancias de una dimensión se les

denomina **miembros**. Por ejemplo, Camisetas, Blusas, Pantalones de Mezclilla y Pantalones Cortos son miembros de la dimensión Producto.

Otro componente del modelo multidimensional es la jerarquía. Una **jerarquía** comprende los niveles de detalle, en los cuales se puede presentar una medida [MAL08]. Las jerarquías se definen sobre las dimensiones asociadas con los cubos y pueden tener distinto número de **niveles** relacionados. En la Figura 10 se creó una jerarquía con los niveles de Categorías y Elementos sobre la dimensión de Producto, mientras que para la dimensión de Territorio de Ventas se tiene una jerarquía con niveles de Ciudad y Estado. La definición de jerarquías y los niveles asociados dependen del análisis que se quiera hacer sobre los datos. Por ejemplo, si en el 2001 se vendieron 2000 blusas en la ciudad de Ottawa y otras 2000 en la ciudad de Toronto, es posible subir un nivel en la jerarquía de Territorio de Ventas y obtener la venta total 4000 blusas para el estado de Ontario en el 2001.

La forma tradicional de presentar el modelo multidimensional a nivel lógico es usando bases de datos relacionales junto con una de sus representaciones, denominadas esquema de estrella y esquema de copo de nieve [GUI12]. El **esquema de estrella** está compuesto por una o más tablas de hechos (Ventas en la Figura 11), cada una ligada a un conjunto de dimensiones (Tienda, Producto, Tiempo y Cliente en la Figura 11). La tabla de hechos contiene las medidas (Ganancia y Cantidad_vendida en la Figura 11) y las llaves foráneas que la relacionan con las tablas de dimensiones.

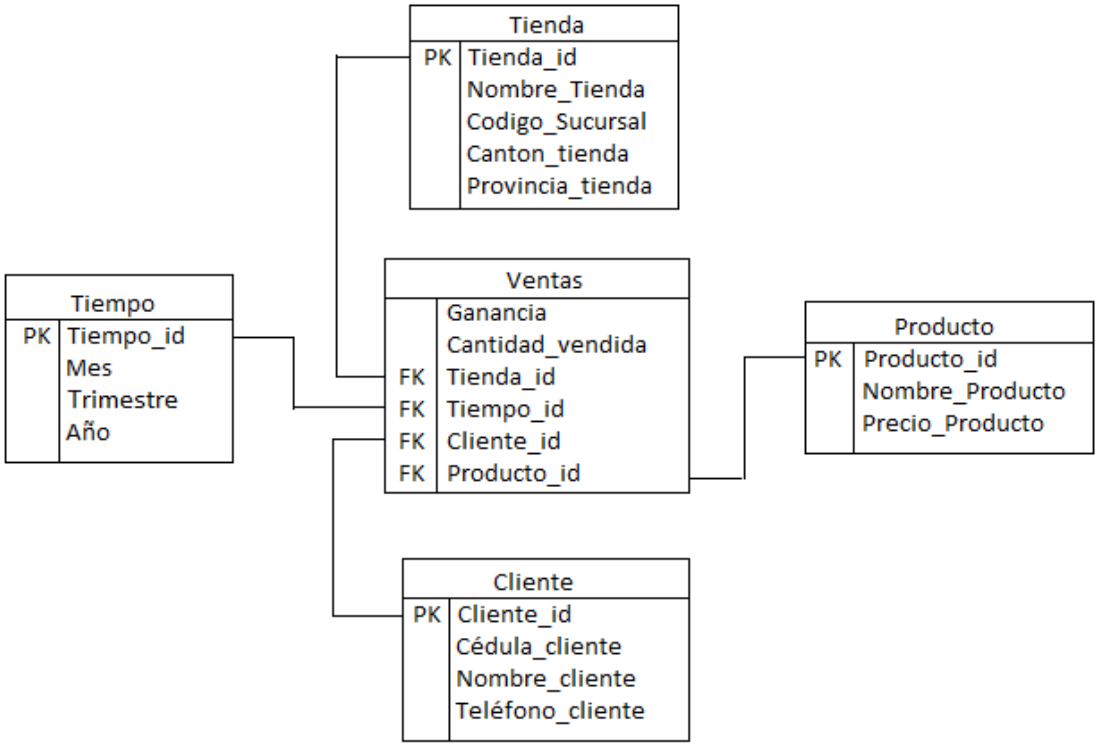


Figura 11. Ejemplo de un esquema de estrella

Las dimensiones en un esquema de estrella no se relacionan directamente entre sí, sino solamente por medio de la tabla de hechos. En caso de que existan jerarquías, estas son incluidas en la misma tabla de dimensiones de forma desnormalizada. Por ejemplo, en la Figura 11, la dimensión Tiempo contiene atributos que pueden representar tres niveles, formando una jerarquía Mes → Trimestre → Año.

El **esquema de copo de nieve** también está compuesto por tablas de hechos asociadas a dimensiones. Pero a diferencia del esquema de estrella, el esquema de copo de nieve se caracteriza por normalizar las tablas de dimensiones para eliminar la redundancia. Al

normalizar, su diagrama se asemeja a un copo de nieve, tal y como se muestra en la Figura 12, donde la jerarquía en la dimensión Tiempo se dividió en tres tablas (una para cada nivel) y la jerarquía en la dimensión Tienda se dividió también en tres tablas: Tienda, Cantón y Provincia.

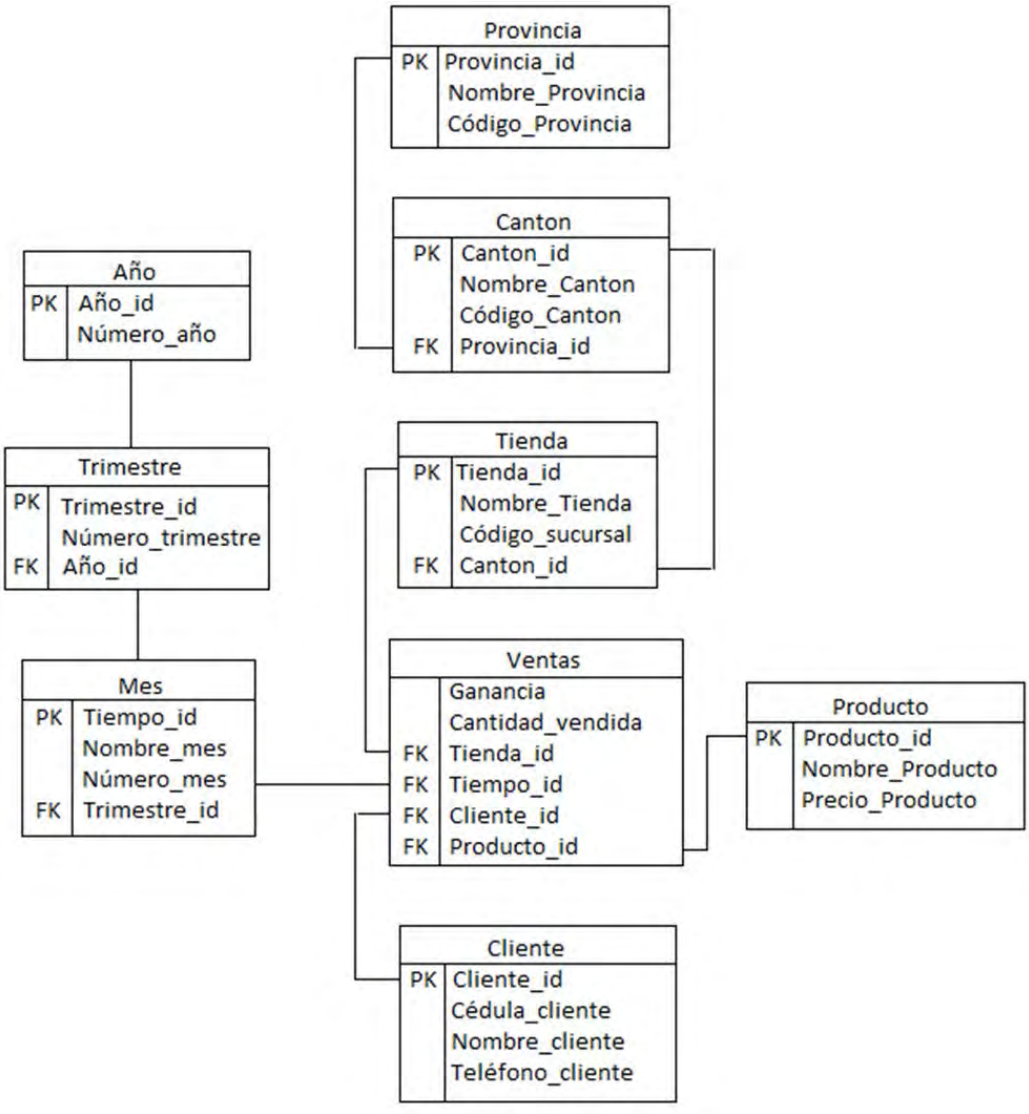


Figura 12. Ejemplo de un esquema de copo de nieve

Almacenes de Datos Espaciales

Se ha comprobado que el 80% de los datos almacenados en bases de datos tiene un componente espacial, el cual generalmente se representa como un nombre, por ejemplo, el de una provincia, una región o un país [MAL08]. Sin embargo, ese tipo de representación no brinda opciones adicionales para el análisis. Por lo tanto, surgió la necesidad de incorporar datos espaciales en el proceso de BI, de modo que, a través de la visualización de resultados en mapas, se convirtieran en una ayuda para el descubrimiento de patrones y pudieran enriquecer el proceso de observación. Como consecuencia, fue imperioso extender la funcionalidad de los almacenes de datos para permitir la inclusión de datos espaciales, dando así origen al concepto de almacén de datos espacial.

Un **Almacén de Datos Espacial** (SDW - *Spatial Data Warehouse*) es aquel que incorpora un elemento anexo para permitir la representación de objetos espaciales y la posterior ejecución de operaciones sobre ellos. Un **objeto espacial** corresponde a una entidad del mundo real para la cual una aplicación necesita almacenar características espaciales, por ejemplo, una Provincia. Este objeto está formado por un componente descriptivo y un componente espacial. El componente descriptivo contiene características del objeto espacial y se representa mediante tipos de datos convencionales como string, int y date; por ejemplo, el objeto Provincia puede ser descrito por un nombre y un código. Por su parte, el componente espacial contiene la geometría.

Las **geometrías** son un conjunto de puntos descritos a través de coordenadas de latitud y longitud, que sirven para representar diferentes objetos espaciales [MAL08]. Como se muestra en la Figura 13, existen diferentes tipos de geometrías, por ejemplo:

- **Punto.** Denota una locación singular en el espacio. Se puede utilizar para representar, por ejemplo, una tienda.
- **Multipunto**. Conjunto de puntos. Puede utilizarse para representar casas en un pueblo, por ejemplo.
- **Línea**. Geometría de una dimensión constituida por un conjunto de puntos conectados que definen segmentos de recta o curva. Puede utilizarse para representar una carretera.
- **Multilínea**. Conjunto de líneas. Puede usarse para representar un sistema de carreteras.
- **Polígono**. Geometría de dos dimensiones que denota un conjunto de puntos conectados, formando una superficie. Se puede usar para representar provincias, cantones y distritos.
- **Multipolígono**. Conjunto de polígonos. Puede utilizarse en el caso de regiones cuya geometría esté formada por más de un polígono.

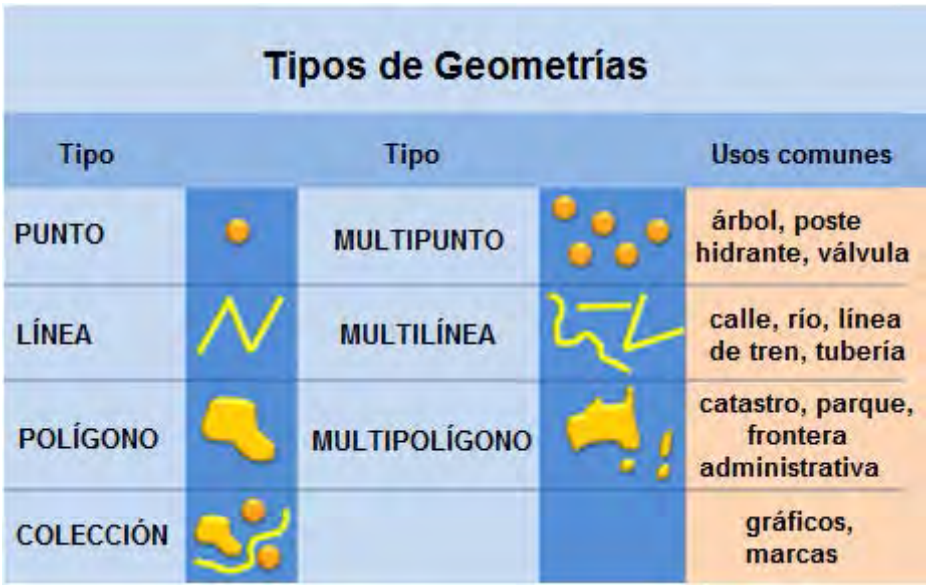


Figura 13. Tipos de geometría [LEA12]

Al trabajar con geometrías, generalmente se especifica un **Identificador de Referencia Espacial** (SRID – *Spatial Reference Identifier*). La importancia de seleccionar el SRID correcto radica en que contiene todos los metadatos sobre el sistema de coordenadas utilizado al recolectar los datos espaciales. Esto hace posible el trazado correcto del dibujo de los mapas y la obtención de resultados acertados al ejecutar operaciones —como el cálculo de área y perímetro— sobre las geometrías.

Físicamente, las geometrías pueden estar almacenadas en *shapefiles*. ***Shapefile*** es un formato de datos desarrollado por el Instituto de Investigación en Sistemas Ambientales (ESRI - *Environmental Systems Research Institute*) que utiliza un vector de coordenadas para almacenar geometrías no topológicas y sus atributos [ENV13]. Gracias a sus ventajas, como un

bajo consumo de espacio en disco y una alta velocidad de trazado [ENV13], estos archivos son utilizados en gran medida por aplicaciones que involucran el despliegue de mapas.

Al igual que un almacén de datos convencional, un SDW contiene tablas de hechos, medidas, dimensiones, jerarquías y niveles. La diferencia radica en que sus elementos pueden representarse mediante geometrías, por lo que se le incorpora el componente espacial, creando así:

- **Niveles espaciales**. Contienen atributos espaciales. Por ejemplo, un distrito con su geometría asociada.
- **Jerarquías espaciales**. Contienen niveles espaciales. Por ejemplo: distrito, cantón y provincia, cada uno asociado a su geometría.
- **Dimensiones espaciales**. Contienen jerarquías espaciales.
- **Medidas espaciales**. Son representadas por una geometría como, por ejemplo, la ubicación de un accidente.

La implementación de un SDW requiere la creación de tres esquemas: uno conceptual, otro lógico y un último físico [MAL08]. Para el desarrollo del esquema conceptual se puede utilizar el modelo *MultiDim* [MAL08], que permite representar elementos espaciales y no espaciales de forma conjunta. Además, incluye las reglas de mapeo a nivel lógico, lo cual facilita su futura implementación. El uso de *MultiDim* demanda, a nivel de implementación, un **Sistema de Administración de Base de Datos** (*Database Management System* – DBMS) que posea una

extensión para soportar los tipos de datos espaciales, con el fin de que permita almacenar los objetos espaciales y, a la vez, preserve las funcionalidades típicas de un DBMS.

3. EXTRACCIÓN, TRANSFORMACIÓN Y CARGA

Debido a que los almacenes de datos se alimentan de distintas fuentes, los datos necesitan pasar por un proceso de extracción y transformación en forma previa a su inserción en el DW. Este proceso es conocido como **Extracción, Transformación y Carga** (ETL - *Extraction, Transformation and Load*) y constituye la capa *back-end* en la arquitectura de un almacén de datos, tal y como se muestra en la Figura 9.

Extracción corresponde a la primera etapa, que consiste en la lectura y copia —temporal— de los datos relevantes para el almacén. En caso de ser necesario, para resolver problemas de interoperabilidad entre las diferentes fuentes de los datos, se pueden utilizar interfaces como ODBC (*Open Data-base Connectivity,*), OLEDB (*Open Linking and Embedding for Databases*) y JDBC (*Java Database Connectivity*), entre otras.

Una vez que los datos se encuentran en el s*taging area*, se procede con la etapa de transformación, que consiste en la aplicación de una serie de cambios a los datos, entre ellos [INM05, KIM11]:

- Limpieza: correcciones ortográficas, resolución sobre elementos faltantes, verificación de consistencia de datos, asignación de valores por defecto (en caso de ausencia de algún dato), entre otros.

- Combinación de datos de distintas fuentes.
- Eliminación de duplicados.
- Asignación de llaves dentro del almacén.

La última etapa del proceso de ETL es la carga de los datos del *staging area* al almacén de datos. Esta fase puede llevarse a cabo insertando los registros uno a uno o utilizando alguna herramienta de carga masiva de datos [INM05].

4. PROCESAMIENTO ANALÍTICO EN LÍNEA

Para realizar operaciones sobre los datos que se encuentran en el almacén, se requiere un servidor que posibilite el **Procesamiento Analítico en Línea** (OLAP - *On-line Analytical Processing*). OLAP es una categoría de software cuyo objetivo es proporcionar un medio para la rápida exploración y el análisis de datos. No sólo posibilita la ejecución de una consulta sobre un conjunto de datos, sino que también permite elegir la forma en que los resultados de esa consulta son visualizados. Además, posibilita la generación y ejecución de nuevas consultas a partir de los cambios que se realicen sobre los resultados desplegados.

OLAP se basa en conceptos del modelo multidimensional presentado anteriormente. La Figura 14[3] muestra un modelo abstracto de un cubo de datos OLAP llamado Negocio, el cual tiene tres dimensiones (Territorio de Ventas, Producto y Tiempo) y una medida (Ventas).

[3] Esta figura presenta el mismo ejemplo de la Figura 10, con el fin de facilitar la lectura.

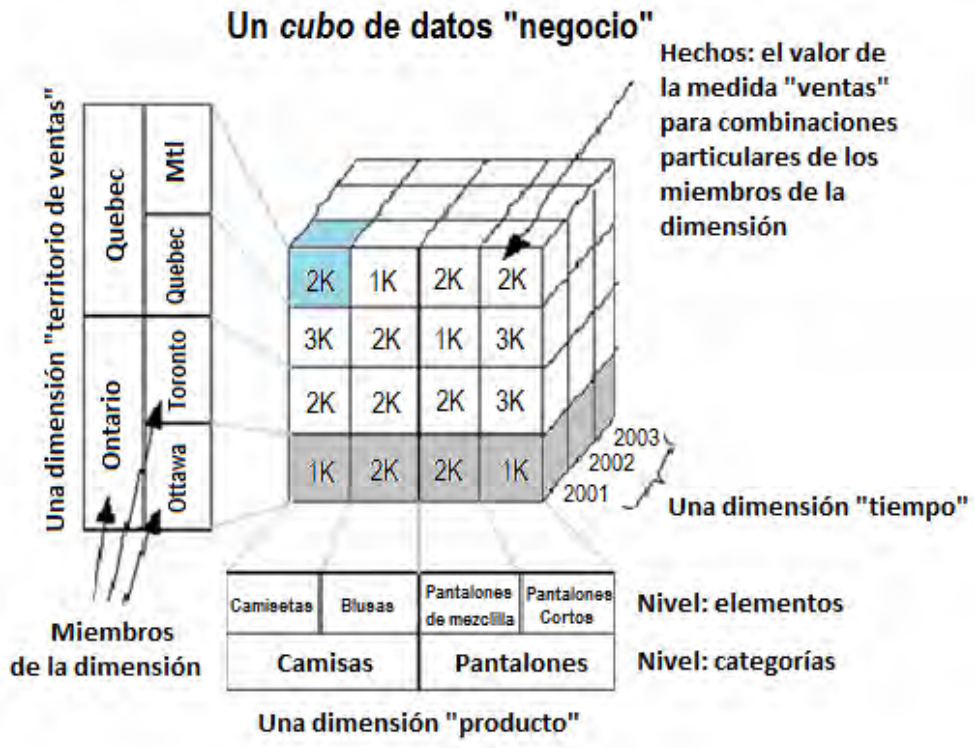


Figura 14. Cubo de datos [RIV03]

La especificidad de las características comunes entre los datos determina la cantidad de niveles en una jerarquía. Así, los miembros de una dimensión deben ser definidos de la forma más específica posible, con el objetivo de que se puedan aprovechar los beneficios de OLAP al agruparlos y desagruparlos. Por ejemplo, en la dimensión Producto de la Figura 14, los miembros del nivel Elementos (Pantalones de Mezclilla y Pantalones Cortos) conforman la definición más específica de Producto y son agrupados para construir un miembro del nivel Categorías (Pantalones). Es decir, los miembros de un nivel de la jerarquía se agrupan para formar un miembro del nivel inmediatamente superior; a esta acción de manipulación del cubo se denomina **agrupación de miembros**.

ADITIVIDAD DE MEDIDAS

En una jerarquía, las medidas están asociadas a los miembros del nivel que presenta un mayor grado de detalle (el más específico). No obstante, los miembros pertenecientes a otros niveles también necesitan que se les asocie un valor para la medida en cuestión. Por ejemplo, en la Figura 14 se podría consultar el valor de la medida Ventas para el miembro Pantalones. En casos como este, el valor de la medida para un miembro puede calcularse a partir de los miembros que lo componen (datos del nivel inferior); para lograrlo, se utiliza una **operación de agregación**.

La agregación es el cálculo de una medida que, utilizando sus valores para los miembros de un nivel, obtiene los resultados correspondientes al nivel inmediatamente superior. En este caso, si la operación de agregación es suma, se calcula en forma automática el valor de la medida para Pantalones como la suma de los valores de Ventas correspondientes a los miembros del nivel inferior que lo componen (Pantalones de Mezclilla y Pantalones Cortos).

Sin embargo, no siempre se puede utilizar la suma como operación de agregación. Esta circunstancia le da origen a tres tipos de medidas [MAL08]:

- **Aditiva**. Es posible sumarla sobre las jerarquías en todas las dimensiones.
- **Semiaditiva**. Tiene sentido lógico sumarla sobre las jerarquías de algunas, pero no todas, las dimensiones.
- **No aditiva**. No existe una dimensión sobre la cual tenga sentido lógico sumarla.

OPERACIONES OLAP

Las operaciones más comunes sobre cubos multidimensionales son: *roll-up, drill-down*, *slice-and-dice* y *pivot*. Las primeras dos generan un cambio en la granularidad —o grado de detalle — del nivel de la jerarquía, mientras que la tercera (*slice-and-dice*) modifica el alcance del cubo [ALB99a] y *pivot* se refiere a la acción de rotar el cubo [ZHE10].

Roll-up

Roll-up permite subir de nivel en la jerarquía. La operación agrupa los miembros de un mismo nivel en clasificaciones más generales correspondientes a un nivel superior y —de forma simultánea— realiza la agregación de medidas. Por ejemplo, en la Figura 14, la dimensión Producto tiene dos miembros en su nivel inferior: Pantalones de Mezclilla y Pantalones Cortos; cuando a uno de ellos se le efectúa *roll-up*, inmediatamente se agrupa con el otro para formar el miembro Pantalones, que pertenece al siguiente nivel hacia arriba. La operación *roll-up* también se encarga de aplicar la función de agregación. Por lo tanto, se suman los valores de la medida Venta que están presentes en Pantalones Cortos (9000) y Pantalones de Mezclilla (7000), tal como se muestra en la Figura 14. El resultado obtenido (16000) se le asigna al miembro Pantalones del nivel Categorías.

Drill-down

Drill-down es la operación opuesta a *roll-up*. Se utiliza para navegar desde el nivel superior —el más general— hacia el nivel inferior —el más detallado— de la jerarquía. Esta operación

expande uno o más miembros específicos de una dimensión del cubo de datos e implica la desagregación de medidas. En el ejemplo anterior —de la Figura 14— se había obtenido el miembro Pantalones del nivel Categorías. Si se le aplica *drill-down* a Pantalones, este se descompone en sus miembros Pantalones de Mezclilla y Pantalones Cortos del nivel Elementos.

Slice-and-dice

Se denomina ***slice-and-dice*** a la capacidad del sistema para aplicar filtros y seleccionar datos específicos dentro de un cubo. La operación ***slice*** consiste en la selección de un subconjunto de datos a partir de la asignación de *un* valor particular para un miembro de una dimensión, de modo que el resultado contenga solamente los datos que cumplen con el criterio especificado. Por ejemplo, la región de color gris en la Figura 14 identifica el resultado de un *slice* sobre el miembro Ottawa de la dimensión Territorio de Ventas. Cuando esta operación se realiza sobre más de dos dimensiones, se utiliza el término ***dice*** en lugar de *slice*. La región celeste de la Figura 14 muestra un ejemplo de la operación *dice*, que se obtiene al seleccionar un miembro particular de cada una de las tres dimensiones disponibles: Camisetas de la dimensión Producto, 2001 de la dimensión Tiempo y Mtl de la dimensión Territorio de Ventas.

Pivot

Pivot rota los ejes del cubo, lo que permite mostrar distintas presentaciones del mismo conjunto de datos. Por ejemplo, la Tabla 2 contiene los resultados obtenidos al realizar una

consulta sobre las Ventas del producto Pantalones Cortos por cada Territorio de Ventas en cada uno de los años.

Tabla 2. Consulta de prueba, antes de hacer *pivot*

Producto	**Territorio de ventas**	**Tiempo**	**Ventas**
Pantalones Cortos	Ottawa	2001	1000
		2002	6000
		2003	6500
	Toronto	2001	3000
		2002	7500
		2003	8500

Al aplicar la operación *pivot* —mostrada en la Tabla 3— en la consulta anterior, se obtienen las ventas del producto Pantalones Cortos por cada año en cada uno de los elementos de Territorio de Ventas.

Tabla 3. Consulta de prueba, luego de hacer *pivot*

Producto	Tiempo	Territorio de ventas	Ventas
Pantalones Cortos	2001	Ottawa	1000
		Toronto	3000
	2002	Ottawa	6000
		Toronto	7500
	2003	Ottawa	6500
		Toronto	8500

OLAP ESPACIAL

Ante la necesidad de utilizar las representaciones de datos espaciales en OLAP, surge el **Procesamiento Analítico Espacial en Línea** (SOLAP – *Spatial On-line Analytical Processing*). SOLAP es definida como una plataforma visual creada especialmente para soportar la exploración de datos y un rápido análisis espacio-temporal, siguiendo un enfoque multidimensional de niveles disponibles mediante distintas formas de despliegue [RIV01]. Como se mencionó en la sección de la arquitectura de un almacén de datos (capa OLAP), SOLAP expande las funcionalidades y características de ROLAP al incorporar el almacenamiento y manipulación de datos espaciales.

Una solución de este tipo contiene un servidor —o motor— SOLAP (encargado de procesar las consultas) y una herramienta de visualización de resultados, que permite mostrar los resultados simultáneamente a través de gráficos, tablas y mapas. También posibilita la ejecución de operaciones típicas de OLAP sobre los datos resultantes y refleja las variaciones producidas por ellas de manera inmediata en todas sus formas de despliegue.

Tal como se muestra en la Figura 15, una herramienta SOLAP puede manejar dimensiones espaciales de tres categorías [RIV03]:

- **No geométricas** (Figura 15a). No contienen una representación cartográfica asociada a los miembros de la dimensión. Utilizan datos nominales, como nombres de lugares.
- **Geométricas** (Figura 15b). Describen las figuras geométricas referenciadas en un mapa (por ejemplo, los polígonos) en todos los niveles de la dimensión.
- **Mixtas** (Figura 15c). Tienen representaciones de figuras geométricas sólo en algunos niveles.

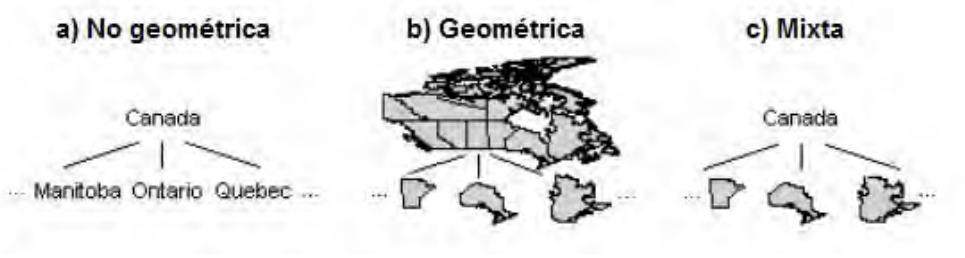


Figura 15. Tipos de dimensiones espaciales en SOLAP [RIV03]

CAPÍTULO III: METODOLOGÍA

GeoCR, la aplicación creada como parte de este proyecto, se diseñó con el objetivo de proveer al CCP con una herramienta que le permitiera analizar datos provenientes de distintas fuentes sobre la plataforma web existente. Para el desarrollo del proyecto, se planteó una arquitectura de tres capas, cada una de las cuales se asoció con una etapa de elaboración del trabajo. Las tres capas se muestran en la Figura 16.

Figura 16. Arquitectura de tres capas de GeoCR

1. CAPA DE DATOS

La primera capa corresponde a la capa de datos, la cual está conformada por todos los elementos relacionados con el SDW y la muestra de datos. Esta capa fue implementada en diferentes etapas, la primera de las cuales correspondió a la discusión con los miembros del CCP sobre los requerimientos de la aplicación y los objetivos que buscaban alcanzar con ella. Seguidamente se obtuvo la muestra de datos que el CCP puso a disposición para el proyecto, la cual fue sometida a un proceso de análisis para determinar las tareas de transformación que se le debían aplicar.

Posteriormente, se llevó a cabo la creación de los esquemas conceptual —creado utilizando el modelo multidimensional— y lógico —basado en el modelo relacional—. Para poder proseguir con el diseño físico, se realizó la escogencia del DBMS a partir de una lista de sistemas *open-source* con capacidad de gestionar datos espaciales. Una vez seleccionado el DBMS, se crearon las tablas usando como base el esquema conceptual. Por último, se tomaron los datos y —luego de aplicarles las transformaciones— se cargaron en el almacén.

2. CAPA LÓGICA

La capa lógica agrupa las tareas relacionadas con el servidor SOLAP y la implementación de los cubos. Para el desarrollo de esta etapa, primero se seleccionó el motor SOLAP *open source*, que sería el encargado de la ejecución de consultas sobre el almacén creado en la capa de datos. Al igual que con el DBMS, la escogencia del motor SOLAP se basó en opciones de

software libre que incluyeran una extensión para manejar datos espaciales, así como en las operaciones OLAP y funciones espaciales que ofrecieran. Posteriormente, se crearon los cubos de datos, para lo cual se realizó el mapeo del esquema multidimensional —creado en la capa de datos— a la implementación física de los cubos, específica para la herramienta seleccionada.

3. CAPA DE PRESENTACIÓN

Esta capa está conformada por las herramientas cliente encargadas de la visualización de resultados en tablas, gráficos y mapas, así como la integración de las mismas en el sitio web del CCP. Para su implementación, primero se realizó una búsqueda de herramientas cliente *open-source* que ofrecieran (1) integración con el motor SOLAP seleccionado, (2) visualización de los resultados en los diferentes formatos requeridos (tablas, gráficos y mapas) y (3) posibilidad de invocar las operaciones OLAP básicas con que cuenta el motor.

Una vez que la herramienta cliente fue seleccionada, se procedió a integrarla con el servidor OLAP, así como a efectuar pruebas que validaran la correcta funcionalidad de la misma. Posteriormente, se realizaron ajustes y personalizaciones a nivel de interfaz, para adaptarla a los gustos y sugerencias de los investigadores del CCP. Luego de que se completaron estas etapas, se prosiguió con la integración de la herramienta cliente al sitio web del CCP.

CAPÍTULO IV: IMPLEMENTACIÓN DEL ALMACÉN DE DATOS ESPACIAL

El desarrollo del almacén de datos espacial consistió en una serie de etapas sucesivas; se inició con la especificación de requerimientos y la selección de la muestra de datos, para luego continuar con la creación del almacén, incluyendo los diseños conceptual, lógico y físico del mismo. Finalmente, se definieron los procesos de ETL para limpiar, integrar, transformar y cargar los datos al nuevo almacén.

1. ESPECIFICACIÓN DE REQUERIMIENTOS

La necesidad planteada consistió en un sistema que permitiera realizar consultas *ad-hoc* sobre los datos de nacimientos, defunciones e incidencia de cáncer disponibles en el CCP. Los objetivos de la institución se extendieron más allá de consultas simples sobre una base de datos, debido a que los usuarios especializados necesitaban de un sistema que integrara los datos existentes y permitiera consultas dinámicas, de fácil generación y que pudieran ser desplegadas en tablas, gráficos y mapas, utilizando diferentes subdivisiones del territorio costarricense. Además, era necesario que todo esto fuera accesible vía web.

La finalidad del CCP en relación con este proyecto, aparte de proveer libre acceso a los datos que ya tenían en su dominio, fue lograr negociaciones con otras instituciones nacionales —

como el INEC— para que obtuvieran beneficios del producto resultante y lo utilizaran en sus estudios. Esto a cambio de que el CCP continuara obteniendo datos estadísticos actualizadosy pudiera ampliar sus posibilidades de análisis.

En cuanto a la especificación de requerimientos, se empleó la combinación de requerimientos orientados al análisis y requerimientos orientados a los datos disponibles [MAL08]. Debido a que el sistema era pequeño, pero con posibilidades de expansión conforme se ampliara su uso, los miembros del CCP puntualizaron los focos de análisis a los que se les debía dar prioridad y brindaron las fuentes de datos en donde se podía encontrar esa información. Los aspectos que se tomaron en cuenta fueron los siguientes:

- Incidencia de los diferentes tipos de cáncer desde diversas perspectivas (edad, género, ubicación geográfica y año).
- Mortalidad ocasionada por determinadas causas, analizándolas desde diferentes puntos de vista (edad, género, localización geográfica y año).
- Análisis de nacimientos según año, género y ubicación geográfica.
- Mortalidad infantil ocasionada por determinadas causas, analizándolas desde diversas perspectivas (horas transcurridas desde el nacimiento, género y localización geográfica).

2. SELECCIÓN DE LA MUESTRA DE DATOS

Los miembros del CCP elaboraron una lista de campos que, de acuerdo con las necesidades de análisis, era necesario incluir dentro de GeoCR. Posteriormente, un investigador encargado

entregó los datos requeridos. La Tabla 4 muestra la lista de los elementos seleccionados; estos fueron extraídos de las bases de datos originales en archivos de texto para facilitar su posterior modificación durante los procesos de ETL y la carga final en el almacén de datos.

Tabla 4. Elementos seleccionados para la muestra de datos

Elementos generales	Descripción
Sexo	Género de los individuos
Edad	Edades agrupadas en conjuntos de cada 5, 10, 15, 20 o más años según el objetivo de análisis
Edad Infantil	Edades de los infantes listadas en rangos de horas
Causa de defunción	Lista de causas de muerte de los individuos
Distritos	Definición de los distritos de Costa Rica
Regiones del CCP	Definición de las regiones de análisis propias del CCP
Regiones de la CCSS	Definición de las regiones y áreas de salud definidas por la CCSS
Regiones del INEC	Definición de las regiones y subregiones de planificación definidas por el INEC
Pertenencia al GAM	Definición del Gran Área Metropolitana
Pertenencia a zona rural	Definición de las zonas rural y urbanas del país
Población	Población asociada cada distrito.
Cáncer	Casos de cáncer registrados desde 1981 hasta el 2005
Defunciones	Defunciones registradas desde 1964 hasta el 2010
Nacimientos	Nacimientos registrados desde 1986 hasta el 2011
Defunciones infantiles	Defunciones infantiles registradas desde 1986 hasta el 2011

3. CREACIÓN DEL ALMACÉN DE DATOS ESPACIAL

Tomando como base los requerimientos descritos en la sección 1 de este capítulo, se crearon los esquemas necesarios: uno conceptual, otro lógico y un último físico.

DISEÑO CONCEPTUAL

El esquema conceptual permite la representación de los requerimientos de datos en una forma clara, concisa y de fácil comprensión para el usuario. Para el trabajo en cuestión, se utilizó el modelo *MultiDim* [MAL08], el cual permite la representación multidimensional de datos convencionales y espaciales dentro del mismo esquema, tal y como se muestra en la Figura 17.

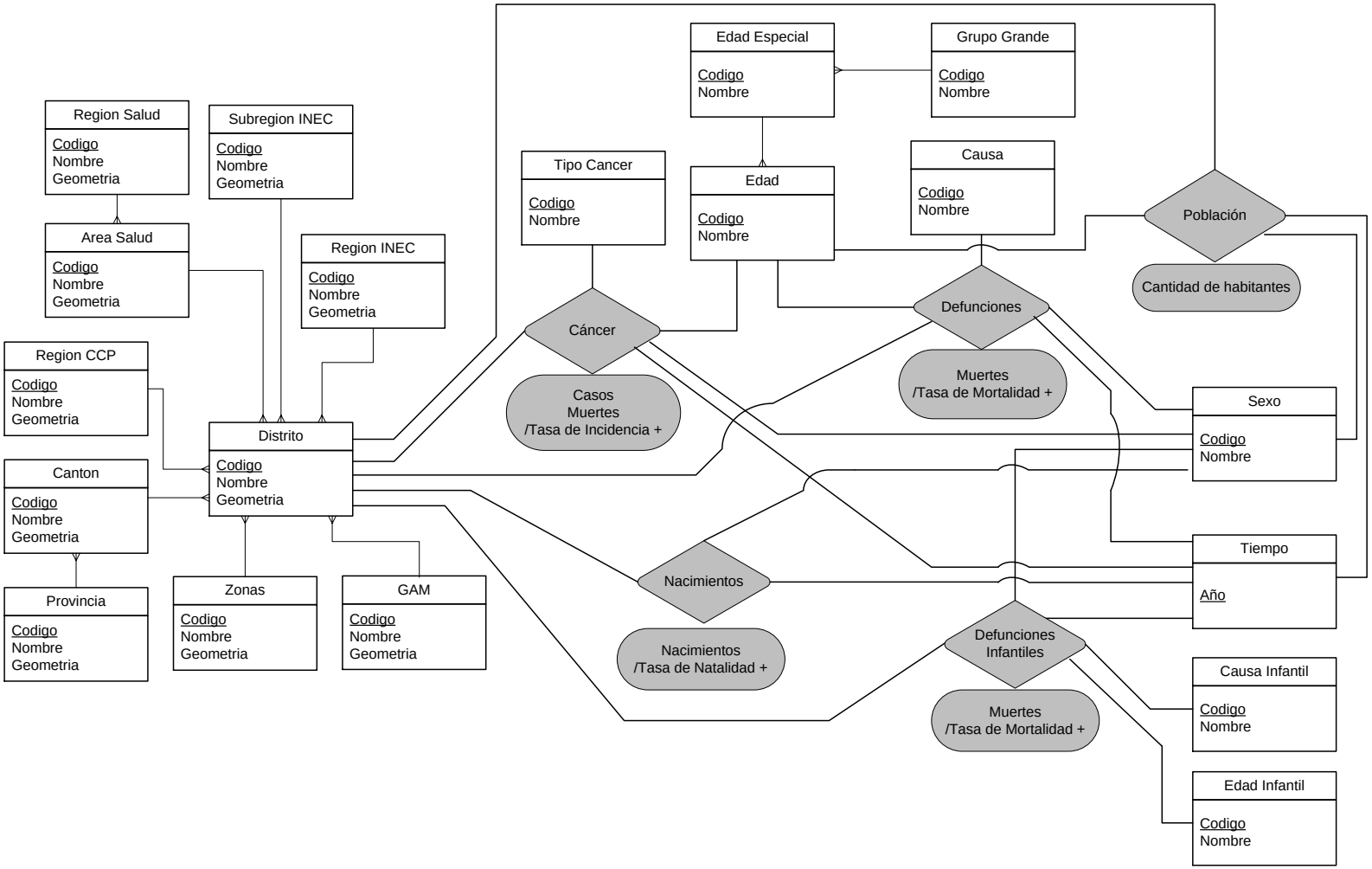


Figura 17. Esquema conceptual de GeoCR

Especificaciones detalladas

Dimensiones

Las dimensiones se asocian a las relaciones factuales y, de acuerdo a sus características, pueden ser de un solo nivel o contener jerarquías. En el esquema de la Figura 17, las dimensiones y los niveles de sus jerarquías están representados por rectángulos. En el caso de que se utilice una jerarquía, el nombre asignado a la dimensión que la contiene equivale al nombre del nivel de menor granularidad. Por ejemplo, en GeoCR, la dimensión que contiene a la jerarquía con los niveles Provincia, Cantón y Distrito es llamada Distritos. A pesar de que algunas dimensiones y niveles de las jerarquías contienen únicamente uno o dos atributos, durante la implementación no necesariamente se mapea cada dimensión o nivel a una tabla separada. El objetivo de esta representación a nivel conceptual es facilitar la comprensión sobre la existencia de diferentes posibilidades de análisis.

La Tabla 5 muestra las características de las dimensiones que contienen jerarquías en el proyecto, mientras que la Tabla 6 contiene las dimensiones de un solo nivel con que cuenta GeoCR. Es importante aclarar que, dado que las subregiones del INEC no son subdivisiones de las regiones del INEC, se creó la jerarquía Subregiones INEC por aparte y no como un nivel de la jerarquía Regiones INEC. Además, la dimensión Edad Infantil no se incluyó dentro de la dimensión Edad porque su granularidad es menor: sus miembros son rangos de horas que tienen como límite alrededor de un año de edad (desde “menos de una hora” hasta “menos de once meses cumplidos”), mientras que el nivel más desagregado de la dimensión Edad está compuesto por grupos de cinco años.

Tabla 5. Dimensiones con jerarquías en GeoCR

Nombre	Contenido	Atributo llave	Atributos descriptivos	Jerarquía	Niveles de la jerarquía	Ejemplo de la jerarquía
Distritos	División territorial de Costa Rica	Código del distrito	Nombre del distrito	Distritos	Provincia, Cantón y Distrito.	El distrito 10203 San Rafael, pertenece al cantón 102 Escazú que a su vez forma parte de la provincia 1 San José.
				Regiones CCSS	Región de salud, Área de salud y Distrito.	El distrito 10203 San Rafael, pertenece al área de salud 610 Escazú, que forma parte de la región de salud 2 Central Sur.
				Regiones CCP	Región del CCP y Distrito.	El distrito 10203 San Rafael, pertenece a la región 5 Rural Bajura.
				Regiones INEC	Región del INEC y Distrito.	El distrito 10203 San Rafael, pertenece a la región 1 Área Metropolitana.
				Subregiones INEC	Subregión del INEC y Distrito.	El distrito 10203 San Rafael, pertenece a la subregión 1 San José.
				Zonas	Zona y Distrito.	El distrito 10203 San Rafael, pertenece a la zona 1 Urbana.
				GAM	Área y Distrito.	El distrito 10203 San Rafael, pertenece al área dentro de la GAM.

Edad	Grupo de edad	Código de edad	Nombre del grupo de edad (en grupos de 5 años)	Edades	Edad grande, Edad especial y Edad.	La edad 40-44 años pertenece al grupo de edad especial 15-44, que a su vez forma parte del grupo de edad grande 15-64.

Tabla 6. Dimensiones de un solo nivel en GeoCR

Nombre	**Contenido**	**Atributo llave**	**Atributos descriptivos**
Tiempo	Año	Año	
Sexo	Género	Código del género	Nombre del género
Tipo	Tipo de cáncer	Código del tipo	Nombre del tipo de cáncer
Causa	Causa de defunción	Código de la causa	Nombre de la causa de defunción
Causa infantil	Causa de defunción infantil	Código de la causa	Nombre de la causa de defunción
Edad Infantil	Edad de infantes	Código de edad	Nombre del grupo de edad (en rangos de horas cumplidas)

Relaciones factuales

En GeoCR existen focos de análisis —conocidos como relaciones factuales— referentes a cáncer, defunciones, nacimientos, defunciones infantiles y población. Dentro del esquema conceptual, los enfoques son representados mediante rombos grises, tal como se puede apreciar en la Figura 17. Además, la Tabla 7 presenta la descripción de los mismos, que contienen medidas —también descritas en la tabla— con granularidad determinada por las dimensiones participantes en la relación factual.

Tabla 7. Relaciones factuales de GeoCR

Nombre	Descripción	Medidas
Cáncer	Permite el análisis sobre la incidencia de distintos tipos de cáncer a través de las diferentes dimensiones (Distrito, Edad, Sexo, Tiempo) y jerarquías.	• Casos: cantidad de casos de cáncer. Es una medida aditiva. • Muertes: cantidad de defunciones causadas por cáncer. Es una medida aditiva. • Tasa de incidencia: tasa de incidencia de cáncer calculada para cada 100000 habitantes. Es una medida no aditiva. • Tasa de mortalidad: tasa de mortalidad de cáncer calculada para cada 10000 habitantes. Es una medida no aditiva.
Defunciones	Permite el análisis sobre la mortalidad originada por distintas causas a través de las diferentes dimensiones (Distrito,	• Muertes: cantidad de defunciones. Es una medida aditiva. • Tasa de mortalidad: tasa de mortalidad calculada para cada

	Edad, Sexo, Tiempo) y jerarquías.	10000 habitantes. Es una medida no aditiva.
Nacimientos	Permite el análisis sobre los nacimientos a través de las diferentes dimensiones (Distrito, Sexo, Tiempo) y jerarquías.	• Nacimientos: cantidad de nacimientos. Es una medida aditiva. • Tasa de natalidad: tasa de natalidad calculada para cada 1000 habitantes. Es una medida no aditiva.
Defunciones infantiles	Permite el análisis sobre la mortalidad de niños menores a un año de edad cumplida, originada por diferentes causas a través de las diferentes dimensiones (Distrito, Sexo, Edad Infantil y Tiempo) y jerarquías.	• Defunciones: cantidad de defunciones. Es una medida aditiva. • Tasa de mortalidad infantil: tasa de mortalidad infantil calculada para cada 1000 nacimientos. Es una medida no aditiva.
Población	Permite el análisis sobre la población a través de las diferentes dimensiones (Distrito, Sexo, Edad y Tiempo) y jerarquías.	• Cantidad de habitantes: cantidad de habitantes. Es una medida aditiva.

Cabe resaltar que Defunciones Infantiles no se incluyó dentro de la relación factual Defunciones porque está asociada a una dimensión de edades que maneja una granularidad diferente (edad en horas) que la ligada a Defunciones (edad en años). Al hacerse esta división, se decidió separar también la dimensión Causa Infantil, porque sus miembros son específicos para Defunciones Infantiles.

De igual forma, la relación factual Población fue creada porque era necesario tener una medida de cantidad de habitantes que no estuviese ligada a los tipos de cáncer y causas de defunciones. Esto porque no es semánticamente correcto analizar una medida Cantidad de habitantes por una causa de muerte; es decir, es posible afirmar que existe una cantidad de habitantes para un distrito, pero no es posible aseverar que existe una cantidad de habitantes para una causa de muerte. Así, para poder realizar correctamente los cálculos de las tasas de incidencia, mortalidad y natalidad, se utiliza la medida Cantidad de habitantes, que está ligada únicamente a las dimensiones de Distrito, Edad, Sexo y Tiempo.

DISEÑO LÓGICO

El diseño lógico se enfoca en la transformación del esquema conceptual multidimensional a un esquema lógico que pueda ser utilizado para implementar en DBMS; por ejemplo, en un modelo relacional. Las siguientes reglas permiten realizar el mapeo del modelo *MultiDim* al modelo lógico relacional [MAL08]:

1. Para la representación de las jerarquías se puede utilizar una de las dos opciones siguientes:
    a. **Tablas normalizadas o estructura de copo de nieve**. Cada nivel se representa en una tabla separada que contiene los atributos del nivel y su llave. En caso de relaciones padre-hijo, la tabla del nivel más bajo contiene una llave externa (foránea) del nivel padre.
    b. **Tablas desnormalizadas o planas**. La llave y los atributos descriptivos de todos los niveles se incluyen en una única tabla.

2. Una relación factual corresponde a una tabla que incluye las llaves primarias de los niveles participantes. Adicionalmente, cada medida se mapea a un atributo de esta tabla.

El mapeo de niveles espaciales es similar al descrito para los niveles convencionales, pero con la incorporación del atributo que representa la geometría. En caso de que existan jerarquías, se aplican las mismas opciones de tablas normalizadas o desnormalizadas. Para representar las jerarquías incluidas en el proyecto, se eligió la opción de tablas desnormalizadas. Esta decisión estuvo determinada por la necesidad de minimizar la cantidad de *joins*, especialmente entre las tablas que contienen atributos complejos tales como las geometrías.

Con el fin de evitar tener relaciones separadas que afecten el desempeño, las dimensiones con uno o dos atributos (como Sexo y Tiempo) se incluyeron como atributos en las tablas de hechos para cáncer, defunciones y nacimientos; esto se conoce como degeneración de dimensiones o dimensiones de hechos [MAL08]. Las consideraciones mencionadas se pueden apreciar en la Figura 18, donde se muestra el modelo lógico final.

Adicionalmente, cabe resaltar que no todas las medidas pudieron ser mapeadas a atributos de una tabla de hechos. En este caso particular, las tasas de incidencia, mortalidad, mortalidad infantil y natalidad son medidas calculadas no aditivas y, por lo tanto, fueron definidas durante la creación de los cubos OLAP.

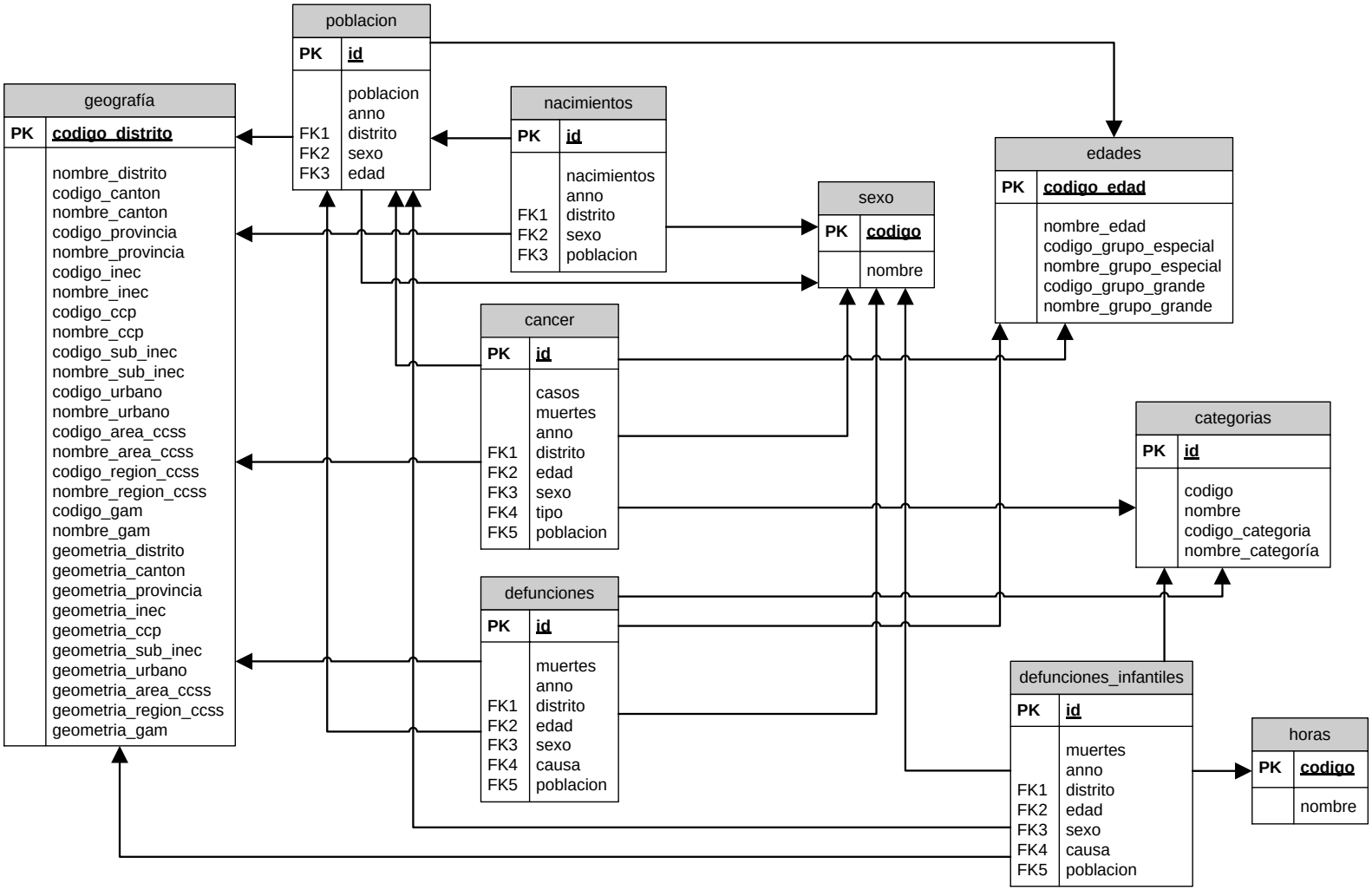


Figura 18. Esquema lógico de GeoCR

DISEÑO FÍSICO

Antes de poder implementar el SDW es necesario seleccionar el DBMS. Actualmente, existen diversos DMBS (Oracle, SQL Server, PostgreSQL, MySQL) que proveen extensiones espaciales para la creación, manipulación y análisis de datos. Sin embargo, siguiendo los requerimientos del CCP —alineados con la legislación costarricense para el uso de software libre sobre el propietario— se seleccionaron MySQL y PostgreSQL como herramientas candidatas. Para elegir entre ambas, se consultó un proyecto de investigación llevado a cabo por la profesora Dra. Elzbieta Malinowski[4] que, tras efectuar un análisis de diferentes DBMS con extensiones espaciales, recomienda PostgreSQL. Adicionalmente, se consideró la comparación de las funcionalidades y capacidades de desempeño que ofrecen los DBMS espaciales realizada en [CHE08]. Con base en los resultados de estas dos investigaciones mencionadas, se seleccionó PostgreSQL junto con su extensión espacial PostGIS.

PostGIS provee dos opciones de tipos de datos espaciales: geography (geografía) y geometry (geometría) [POS04]. La diferencia entre ambos está en que geography brinda una representación esférica de los datos espaciales, mientras que geometry ofrece una representación plana de coordenadas. Actualmente existen menos funciones disponibles para geography que para geometry; tomando esto en cuenta y añadiéndolo a su mayor eficiencia en el cálculo, se escogió geometry como el tipo de dato utilizado en el proyecto. Debido al tamaño del territorio costarricense, la escogencia entre una representación esférica o una plana no

[4] Malinowski, E. Informe final del proyecto de investigación N° 326-B0-120 "Bases de datos espaciales como alternativa para ofrecer el servicio de mapas en un ambiente integrado", ECCI, UCR, 2011.

provoca una notable diferencia en los resultados de los cálculos, como sí ocurriría en territorios más extensos.

PostGIS permite rastrear y reportar los tipos de geometrías usados en la base de datos por medio de dos estructuras [POS04]:

- Spatial_ref_sys. Tabla que contiene todos los sistemas de referencia espacial que pueden ser utilizados en la base de datos.
- Geometry_colums. Vista que contiene la lista de todas las propiedades asociadas con columnas de tipo geometry.

Los pasos para instalar PostgreSQL y su extensión PostGIS se muestran en el Anexo A. Una vez que ambos fueron instalados, se creó el almacén de datos ejecutando los comandos mostrados en el Código 1. Primero, se creó la base de datos (línea 1) y se habilitó el lenguaje PL/pgSQL (línea 2) en la nueva base de datos (esto es necesario para poder ejecutar varias funciones de PostGIS). Posteriormente, se cargaron los objetos y definiciones de funciones propias de PostGIS en el almacén (línea 3); y por último, se cargaron los identificadores y definiciones del sistema de coordenadas en la tabla spatial_ref_sys (línea 4).

Código 1. Comandos para crear almacén de datos en PostGIS [POS04]

1	`createdb almacen_datos_ccp`
2	`createlang plpgsql almacen_datos_ccp`
3	`psql -d almacen_datos_ccp -f /usr/share/postgresql/9.1/contrib/postgis-2.0/postgis.sql`
4	`psql -d almacen_datos_ccp -f /usr/share/postgresql/9.1/contrib/postgis-2.0/spatial_ref_sys.sql`

Código 2. Fragmento de la sentencia para crear la tabla con geometrías en PostGIS

```
1  CREATE TABLE geografia (
2  id SERIAL                    PRIMARY KEY   UNIQUE,
3  codigo_distrito              int4,
4  nombre_distrito              varchar(50),
5  geometria_distrito           geometry(POLYGON,4326),
6  codigo_canton                int4,
7  nombre_canton                varchar(50),
8  geometria_canton             geometry(POLYGON,4326),
9  codigo_provincia             int4,
10 nombre_provincia             varchar(50),
11 geometria_provincia          geometry(POLYGON,4326),
12 codigo_inec                  int4,
13 nombre_inec                  varchar(50),
14 geometria_inec               geometry(POLYGON,4326),
15 …
```

Una vez completada la creación del almacén de datos, se procedió a crear las tablas necesarias. En el caso de las tablas asociadas con dimensiones espaciales (que incluyen geometrías), se especifican las columnas de tipo `geometry` con su respectivo SRID[5] dentro de la sentencia de creación. En GeoCR se utilizó el SRID 4326, que corresponde al **Sistema Geodésico Mundial 1984** (WGS84 - *World Geodetic System 84*) y es el estándar obligatorio, determinado así por el decreto Nº 33797-MJ-MOPT del 30 de marzo del 2007 [PRE14]. Por

[5] Como se indicó en el marco teórico, SRID significa Identificador de Referencia Espacial (*Spatial Reference Identifier*).

ejemplo, el fragmento mostrado en Código 2 es parte de la sentencia utilizada para crear la tabla geografia con sus respectivas columnas de tipo geometry.

Otras tablas —las de las dimensiones no espaciales y las tablas de hechos— se crean con sentencias de SQL más simples, como el fragmento mostrado en el extracto Código 3. El Anexo A contiene los códigos completos de creación de tablas.

Código 3. Fragmento de sentencia para crear una tabla en PostGIS

```
1   CREATE TABLE cancer(
3   distrito      int4,
4   anno          int4,
5   sexo          int4,
7   edad          int4,
8   …
```

4. PROCESOS DE ETL

En la etapa de ETL del proyecto, tomando en cuenta la especificación de requerimientos y la definición de la muestra de datos mencionada en las secciones 1 y 2 de este capítulo, el proceso de extracción de datos fue realizado por los funcionarios de CCP con acceso a las bases de datos existentes en esta institución. En el caso de los datos convencionales, estos fueron recibidos dentro de archivos con formato SQL, los cuales se analizaron para realizar las transformaciones necesarias y posteriormente se cargaron en el SDW. Por otra parte, para los datos espaciales se utilizaron los *shapefiles* provistos por el CCP, a partir de los cuales se extrajeron las geometrías que fueron recibidas en archivos con formato KML. En las secciones

subsiguientes, los procesos de transformación y carga se explican por separado para datos convencionales y datos espaciales.

TRANSFORMACIONES SOBRE DATOS CONVENCIONALES

Los datos convencionales extraídos de las fuentes fueron exportados por miembros del CCP a archivos SQL, donde los valores se encontraban separados por comas y el encabezado del archivo era la sentencia SQL INSERT INTO. Por ejemplo, para los distritos se obtuvo un archivo con el formato mostrado en el extracto Código 4.

Código 4. Formato original de los archivos fuente

1	INSERT INTO geografia (codigo_distrito, nombre_distrito,codigo_canton,nombre_canton,codigo_provincia,nombre_provincia,codigo_inec,nombre_inec,codigo_ccp,nombre_ccp,codigo_sub_inec,nombre_sub_inec,codigo_urbano,nombre_urbano,codigo_area_ccss,nombre_area_ccss,codigo_region_ccss,nombre_region_ccss,codigo_gam,nombre_gam)
2	VALUES (10101,'10101 CARMEN',101,'101 SAN JOSE',1,'1 SAN JOSE',1,'1 AREA METROPOLITANA',1,'1 METRO SAN JOSE',1,'1 SAN JOSE',3,'3 RURAL CONCENTRADO',626,'626 CATEDRAL NORESTE',2,'2 CENTRAL SUR',1,'DENTRO'),
3	(10102,'10102 MERCED',101,'101 SAN JOSE',1,'1 SAN JOSE',1,'1 AREA METROPOLITANA',1,'1 METRO SAN JOSE',1,'1 SAN JOSE',3,'3 RURAL CONCENTRADO',614,'614 MATA REDONDA-HOSPITAL',2,'2 CENTRAL SUR',1,'DENTRO'),
4	…;

Antes de realizar cualquier cambio, se analizó una muestra de datos para definir cuáles transformaciones se deberían aplicar de forma previa a la carga en el SDW. Cada uno de los aspectos analizados y las transformaciones correspondientes se listan a continuación.

Aspecto 1:

Se encontraron nombres de campos —como provincias, cantones, tipo de cáncer y otros— demarcados por comillas dobles en el CSV.

Transformación:

Considerando la sintaxis requerida por SQL para la inserción de datos tipo *varchar* se reemplazaron las comillas dobles (“) por comillas simples (‘) utilizando la función de reemplazo incorporada en editores de texto.

Aspecto 2:

Algunos datos originales estaban incompletos. Por ejemplo, existían entradas en los registros de cáncer en los que se conocía la provincia y el cantón, pero no el distrito. También se presentaron casos en que se conocía la provincia, pero no el cantón y el distrito.

Transformación:

Los datos originales contenían los valores 0, 8 o 9, donde 0 y 9 indicaban valores desconocidos y 8 indicaba extranjeros. Luego de consultar con los miembros del CCP se concluyó que no tenía sentido incluir datos con esas características en el análisis, porque, al no estar relacionados con un distrito, sería imposible representarlos sobre el mapa. Considerando esa limitación, se decidió eliminar esos registros.

Aspecto 3:

Los datos originales estaban incompletos en cuanto a las etiquetas de ciertas divisiones territoriales, como las regiones y subregiones del INEC, la CCSS y el CCP. Adicionalmente, no contaban con la relación entre esas divisiones y los distritos que las conformaban.

Transformación:

Primero, se obtuvieron las etiquetas y códigos de cada una de las regiones. Los datos fueron adquiridos de distintas fuentes:

- Regiones y áreas de salud fueron tomadas de [MOR07, ROS02].
- Regiones del CCP fueron tomadas de [ROS02].
- Regiones y sub-regiones del INEC fueron tomadas de [ROS02].
- Categorías de zonas (rurales y urbanas) fueron tomadas de [ROS02].
- Pertenencia al GAM fue tomada de [ROS02].

Tras analizar los archivos encontrados, se realizó una asociación de los distritos con cada una de las regiones y subregiones a las cuales pertenecen, utilizando las funciones de búsqueda y reemplazo incorporadas en editores de texto.

Aspecto 4:

Los datos extraídos no contenían históricos para los distritos creados recientemente.

Transformación:

Consultando con los miembros del CCP, se procedió a eliminar los registros existentes para los distritos nuevos, como Brunca y El Roble.

Aspecto 5:

Los datos originales no incluían etiquetas para los diferentes grupos de edad.

Transformación:

Los miembros del CCP especificaron los rangos que debían manejar los grupos de edad valiosos para el análisis. Posteriormente, se crearon las etiquetas de las categorías necesarias.

Aspecto 6:

Dentro del archivo SQL que contenía los datos de defunciones infantiles, algunos registros no tenían un valor asignado para la medida Muertes (el campo se encontraba vacío).

Transformación:

Se agregó un cero para indicar que no se encontraron defunciones.

Aspecto 7:

Los caracteres especiales del español —como las tildes— no podían incluirse como parte del valor de una columna dentro de una tabla.

Transformación:

Se cambió la codificación utilizada por la base de datos para permitir tildes.

Aspecto 8:

La nomenclatura de los datos fuente sobre los tipos de cáncer y defunciones debía cambiarse por solicitud del CCP.

Transformación:

El CCP generó un archivo con el mapeo entre los códigos originales y los que se debían insertar en el almacén (ver Anexo E). Para aplicar los cambios se utilizaron funciones de búsqueda y reemplazo del editor de texto, de modo que los valores se ajustaran a los nuevos códigos.

Aspecto 9:

Las causas de mortalidad infantil debían ser distintas a las que existían para el resto de defunciones.

Transformación:

El CCP elaboró una lista con las causas para las defunciones infantiles, que se incluye en el Anexo E. En ella, se indican las etiquetas de esas nuevas causas, junto a los identificadores de las causas originales —las de defunciones— que formarían parte de la nueva clasificación. De esa manera, se crearon las categorías necesarias para la tabla de causas de defunciones infantiles Por ejemplo:

- Las causas 6, 9, 10, 11, 12, 13, 14, 17, 18, 21, 22, 23 y 24 de defunciones pasaron a formar parte de la causa de defunción infantil Residual.
- Las causas 2, 3 y 4 de defunciones se unieron para formar la causa de defunción infantil Resto Infecc.
- Las causas 19 y 20 de defunciones constituirían la causa de defunción infantil Otros accidentes.

TRANSFORMACIONES SOBRE DATOS ESPACIALES

Los datos espaciales corresponden a las geometrías de las diferentes regiones, distritos, cantones y provincias. Originalmente, los datos generados por miembros especializados del CCP contenían imperfecciones; sin embargo, ellos los corrigieron aplicándoles un proceso llamado armonización de nodos, realizado a través de la herramienta MapLogix de MapInfo. La Figura 19 ilustra los beneficios producidos por este método, el cual analiza imperfecciones según el tipo de geometría y toma acciones para simplificar los datos espaciales. Así, la Figura 19a muestra la presencia de puntos y líneas que entorpecen el análisis (cerca de las fronteras de las geometrías), mientras que la Figura 19b ilustra la reducción de estos elementos como producto del proceso de limpieza aplicado.

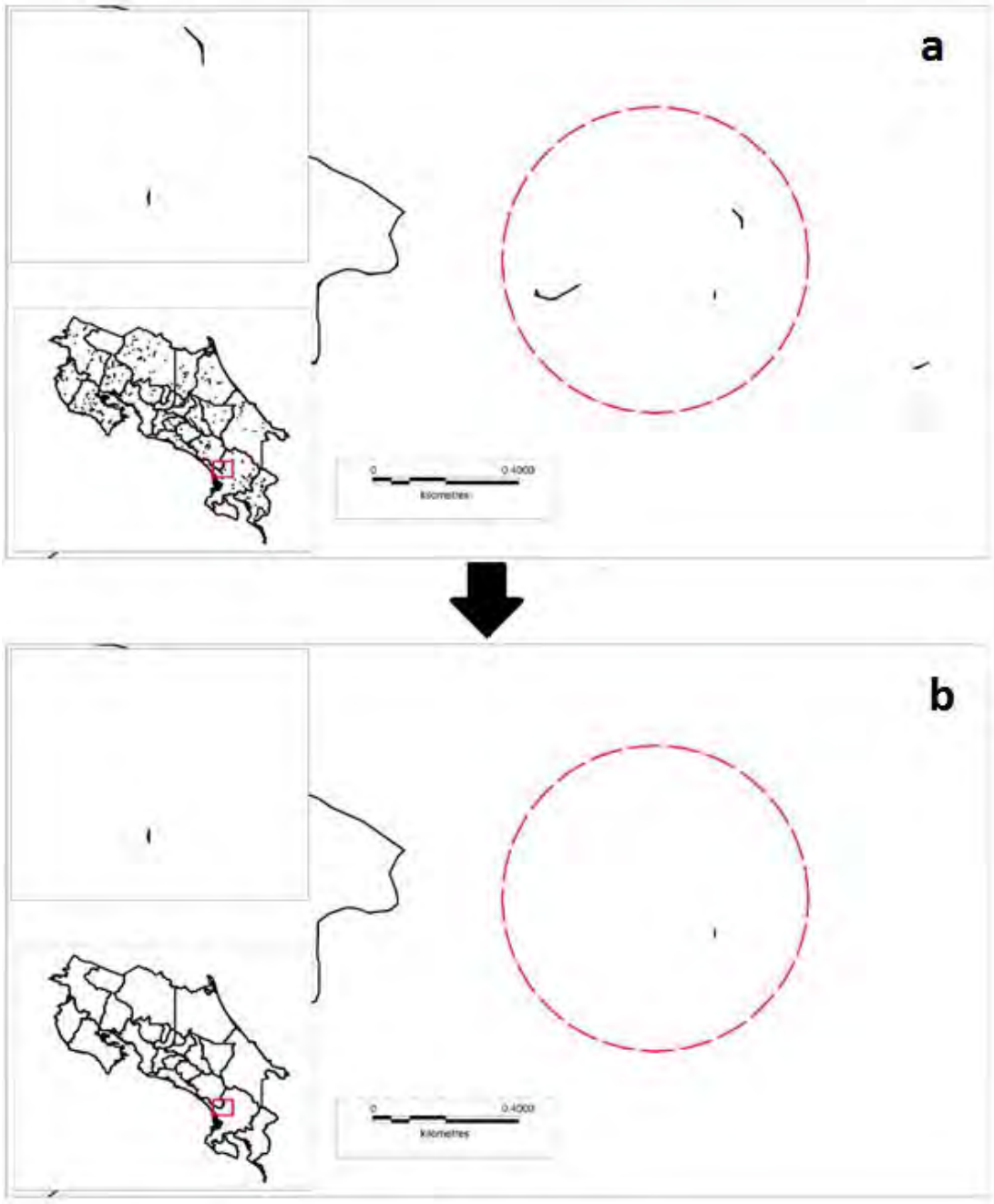


Figura 19. Simplificación de geometrías

a) Geometría con imperfecciones b) Geometría luego de la limpieza

Los datos espaciales simplificados fueron exportados a archivos KML, con el objetivo de incluir el código de la división territorial correspondiente dentro de las etiquetas del fichero. El extracto Código 5 muestra un fragmento de un archivo KML que ilustra la información adicional a la geometría contenida en el archivo.

Código 5. Fragmento de archivo KML con etiquetas y geometría de un cantón

```
1  <Placemark>
2     <ExtendedData><SchemaData schemaUrl="#crcant2_region3">
3        <SimpleData name="Pc">101</SimpleData>
4        <SimpleData name="Canton">SAN JOSE</SimpleData>
5        <SimpleData name="Tasa">12.23</SimpleData>
6        <SimpleData name="Xcentroide">-84.11</SimpleData>
7        <SimpleData name="Ycentroide">9.94</SimpleData>
8        …
9     </SchemaData></ExtendedData>
10       <Polygon><outerBoundaryIs><LinearRing><coordinates>-
             84.089276874061142,9.947232621461852                    -
             84.083203651517195,9.946468480741004…
         </coordinates></LinearRing></outerBoundaryIs></Polygon>
11 </Placemark>
```

Como parte de las transformaciones, los archivos se editaron de manera que quedaran en el formato adecuado para su inserción en PostgreSQL. Las transformaciones aplicadas fueron las siguientes:

- Eliminación de encabezados propios de KML, conservando únicamente el código de la región territorial.
- En el caso de las geometrías que inician con la etiqueta Polygon, se añadieron las funciones ST_Multi y ST_GeomFromKML para transformar la geometría a un formato compatible con PostGIS, tal y como se muestra en el extracto Código 6.

Código 6. Ejemplo de formato aceptado por PostGIS para geometrías

```
1  Select
2  ST_Multi(ST_GeomFromKML('<Polygon><outerBoundaryIs><LinearRing><coordina
   tes>-84.089277,9.947233 …'))
```

- En el caso de las geometrías que inician con la etiqueta MultiGeometry (geometría formada por varios polígonos), se requirió transformar el formato y utilizar las funciones ST_Multi y ST_GeomFromText como lo muestra el extracto Código 7.

Código 7. Ejemplo de formato aceptado por PostGIS para multipolígonos

```
1  Select
2  ST_Multi(ST_GeomFromText('MULTIPOLYGON(((-84.162352 10.577538,… ')))));
```

Al finalizar las transformaciones, se agregaron las etiquetas propias de la sentencia UPDATE de SQL y se cambió la extensión del archivo resultante a SQL. Una vez completada la corrección del formato, el contenido incluyó datos como los mostrados en los extractos Código 8 y Código 9. Ambos fragmentos muestran en la línea 3 la cláusula WHERE con el código de región

correspondiente. Esto fue posible gracias a los encabezados de KML que incluyeron, además de la geometría, el código de la región asociada, tal y como se mostró en Código 5.

Código 8. Inserción de la geometría de tipo polígono en la tabla geografía existente

```
1  UPDATE geografia
2  SET geometria_canton=(Select
   ST_Multi(ST_GeomFromKML('<Polygon><extrude>1</extrude><altitudeMode>cla
   mpToGround</altitudeMode><tessellate>1</tessellate><outerBoundaryIs><Li
   nearRing><coordinates>
         -83.472041,10.290638
         -83.472406,10.290829
         -83.472661,10.290802 …
   </coordinates></LinearRing></outerBoundaryIs></Polygon>')))
3  WHERE codigo_canton = 706;
```

Código 9. Inserción de la geometría de tipo multipolígono en la tabla geografía existente

```
1  UPDATE geografia
2  SET geometria_canton = (Select
   ST_Multi(ST_GeomFromText('MULTIPOLYGON(((-84.162352 10.577538,… ,-
   84.231112 10.168519)))', 4326)))
3  WHERE codigo_canton = 203;
```

CARGA DE DATOS

Los archivos SQL creados en la sección de transformación se utilizaron posteriormente para cargar los datos en el almacén por medio del comando de PostgreSQL:

```
psql -U <nombreUsuario> -d <NombreBD> -a -f <archivoSQL>
```

Una vez ejecutada esa instrucción, se asume que los datos se encuentran debidamente insertados en el almacén de datos y están listos para ser consultados.

CAPÍTULO V: IMPLEMENTACIÓN DE CUBOS SOLAP

Los servidores OLAP acceden al almacén de datos para obtener datos y, a partir de ellos, generan resultados que otras herramientas se encargan de desplegar. En el caso del GeoCR, se recurrió a su utilización para crear cubos OLAP basados en los datos del almacén de datos, solventando así la necesidad de implementar la capa lógica. De acuerdo con la investigación realizada y con los requerimientos inherentes al proyecto (necesidad de incorporar representaciones de diferentes espacios geográficos), sólo se encontró un servidor OLAP *open-source* con la capacidad de manejar datos espaciales: GeoMondrian [SPA22].

1. IMPLEMENTACIÓN CON GEOMONDRIAN

GeoMondrian es una herramienta creada con el objetivo de posibilitar el análisis espacial en Mondrian[6], el servidor OLAP de Pentaho. Como fue desarrollada a partir de Mondrian, heredó su estructura y funcionalidad básica [SPA22], características a las cuales se añadieron dos particularidades: soporte de datos espaciales y funciones específicas para ejecutar sobre estos.

[6] Pentaho Mondrian Project, "Pentaho Mondrian Documentation". [En línea]. Disponible: http://community.pentaho.com/projects/mondrian/ [Último acceso: 2 de febrero, 2014]

DISEÑO BÁSICO DEL ESQUEMA

GeoMondrian construye cubos a partir de esquemas definidos por medio de archivos en formato XML. En ellos se especifican los elementos básicos que debe incluir el análisis, como los cubos, compuestos por las dimensiones, medidas y jerarquías.

Código 10. Esquema simple de GeoMondrian

```
1  <Schema>
2     <Cube name="Cancer">
3          <Table name="cancer_fact"/>
4          <Dimension name="Distritos" foreignKey="codigo_distrito">
5               <Hierarchy hasAll="false" primaryKey="codigo_distrito">
6                    <Table name="geografia"/>
7                    <Level name="Provincia" column="nombre_provincia"
                     type="String" />
8                         <Property name="geom"
                          column="geometria_canton" type="Geometry" />
9                    <Level name="Cantón" column="nombre_canton"
                     type="String" />
10                        <Property name="geom"
                          column="geometria_canton" type="Geometry" />
11                   <Level name="Distrito" column="nombre_distrito"
                     type="String" />
```

```
12                          <Property                          name="geom"
                            column="geometria_canton" type="Geometry" />
13              </Hierarchy>
14          </Dimension>
15          <Measure name="Casos" column="casos" aggregator="sum"/>
16          <Measure name="Muertes" column="muertes" aggregator="sum"/>
17      </Cube>
18  </Schema>
```

El extracto Código 10 muestra un esquema simple en GeoMondrian. En él, se incluye un cubo llamado Cáncer, que a su vez contiene la jerarquía Distrito (con sus niveles Distrito, Cantón, Provincia) como parte de la dimensión Distritos. Los parámetros indicados dentro de la etiqueta que comienza con Level en la definición del nivel son los encargados de señalar su nombre, ubicación (columna de la tabla) y tipo. Por ejemplo, en el Código 10, el componente descriptivo del nivel Provincia (fila 7) es de tipo String y sus instancias están en la columna nombre_provincia. Tanto los componentes descriptivos como las medidas Casos y Muertes definidas en el cubo (filas 15 y 16) se encuentran almacenados en la tabla cancer_fact (fila 3).

Además del componente descriptivo, un nivel de la jerarquía definida en un esquema también puede contener un componente espacial. Para añadirlo se necesita la especificación de una propiedad dentro de la definición del nivel, lo cual se logra al utilizar la etiqueta Property, junto con un nombre, la columna en la tabla y su tipo. Por ejemplo, en el Código 10, el componente espacial del nivel Provincia se llama geom, se encuentra en la columna

geometría_provincia y es de tipo Geometry (fila 8). Los componentes espaciales a los que se hace referencia desde el esquema se encuentran almacenados en la tabla geografía (fila 6).

Además de Cáncer (que se muestra en el ejemplo del Código 10), en el esquema final se definieron de forma similar los cubos Población, Defunciones, Nacimientos y Defunciones infantiles. Todos ellos se utilizaron para crear los cubos virtuales, a los que se hace referencia en una sección posterior. En el Anexo H se presenta el esquema completo que se implementó en GeoCR.

DIMENSIONES COMPARTIDAS

La necesidad de compartir dimensiones entre dos o más cubos está presente en sistemas complejos de DW y sistemas OLAP espaciales. Por ejemplo, si en el esquema del Código 10 se incluyera un nuevo cubo que también utilizara la dimensión Distritos (especificada dentro del cubo Cáncer), esta tendría que ser nuevamente definida dentro del cubo recién agregado. De esa manera, la especificación de una dimensión tendría que realizarse tantas veces como la cantidad de cubos en que se utiliza, creando una repetición innecesaria de definiciones a lo largo del esquema. Para evitar ese inconveniente, GeoMondrian permite compartir dimensiones entre los cubos. El esquema construido para este proyecto contiene dimensiones y jerarquías sobre el territorio nacional, que —como se mostró en la Figura 17— son utilizadas por todos los cubos.

Código 11. Uso de dimensiones compartidas en un esquema de GeoMondrian

```
1   <Schema>
2       <Dimension name="Distritos" foreignKey="codigo_distrito">
3       <!--Misma definición que dimensión Distritos de Código 10-->
4       </Dimension>
5       <Cube name="Cancer">
6             <!--…-->
7             <DimensionUsage   name="Distritos"   source="Distritos"
              foreignKey="distrito"/>
8             <!--…-->
9       </Cube>
10      <Cube name="Defunciones">
11            <!--…-->
12            <DimensionUsage   name="Distritos"   source="Distritos"
              foreignKey="distrito"/>
13            <!--…-->
14      </Cube>
15  </Schema>
```

En GeoMondrian, las dimensiones compartidas deben definirse fuera de los cubos, facilitando así su reutilización y a la vez simplificando la definición del esquema. El extracto Código 11 muestra un ejemplo del uso de dimensiones compartidas, donde la dimensión Distrito se define una sola vez fuera de la definición de los cubos (filas 2-4) y es compartida entre los cubos Cáncer y Defunciones. Estos cubos hacen referencia a esta dimensión utilizando una sentencia con la etiqueta DimensionUsage, a través de la relación entre la llave foránea distrito

de la tabla de hechos (filas 7 y 12) y la llave primaria codigo_distrito, especificada en la definición de la dimensión (fila 2).

MEDIDAS CALCULADAS

Las medidas calculadas son aquellas que no están explícitamente en el almacén de datos, pero que pueden ser calculadas a partir de otras existentes; es decir, sus valores no están en una columna de la tabla de hechos, sino que son el resultado de la aplicación de una fórmula. Por ejemplo, en el cubo Cáncer podría agregarse una medida calculada para obtener la cantidad de sobrevivientes de cáncer, mediante la sustracción de la cantidad de Muertes al número de Casos. No obstante, es importante aclarar que el ejemplo de la medida Sobrevivientes no es aplicable en este proyecto, puesto que los datos de Casos que se manejan son los reportados cada año y, para que la medida Sobrevivientes cumpliera su propósito, se requeriría que estos fueran acumulativos (el total de casos registrados en el año anterior sumados a los nuevos casos reportados en el año actual).

El Código 12 muestra la definición de la medida calculada Tasa de incidencia empleada en GeoCR. Su especificación en el esquema se hace por medio de la cláusula CalculatedMember, donde el parámetro name se utiliza para especificar el nombre de la medida calculada, mientras que el parámetro dimension con el valor Measures indica que la nueva medida forma parte del grupo de medidas disponibles en el cubo. Asimismo, la cláusula Formula es necesaria para definir el cálculo de la medida, que se presenta en lenguaje **Expresiones Multidimensionales** (MDX - *MultiDimensional eXpressions*). En el caso de la medida Tasa de incidencia, se realiza la división de los Casos entre Cantidad de habitantes y se multiplica el

resultado por 100.000, con lo que se obtiene la cantidad de casos de cáncer por cada cien mil habitantes.

Código 12. Uso de medida calculada en un esquema de GeoMondrian

```
1   <CalculatedMember name="Tasa de incidencia " dimension="Measures">
2       <Formula>
            <!--Verificación de que el valor para "Cantidad de
            habitantes" sea mayor a cero y diferente de nulo-->
            [Measures].[Casos]/(Aggregate({[Tipo Cáncer].[All Tipos]},
            [Measures].[Cantidad de habitantes]))*100000
        </Formula>
3   </CalculatedMember>
```

Al momento de realizar una consulta que incluye una medida calculada, GeoMondrian obtiene el valor de cada una de las medidas dentro de la fórmula y les aplica el filtro especificado. Este comportamiento puede generar problemas cuando se trabaja con medidas pertenecientes a cubos con distinta granularidad. Por ejemplo, si se definiera una medida Tasa como Casos entre Cantidad de habitantes y se realizara una consulta de su valor para Leucemia en el Distrito A, la herramienta trataría de obtener (1) el valor de Casos de Leucemia en el Distrito A y (2) el valor de Cantidad de habitantes de Leucemia en el Distrito A. Sin embargo, al no poder agregar Cantidad de habitantes sobre la dimensión Tipo Cáncer, el segundo cálculo fallaría y, por lo tanto, también lo haría el cálculo de Tasa. Este problema surge porque el cubo del que se origina Casos (Cáncer) tiene un menor nivel de granularidad que el que contiene Cantidad de habitantes (Población). Como se puede constatar en el esquema conceptual de la Figura 17,

el cubo Cáncer cuenta con una dimensión más que el cubo Población: Tipo Cáncer; etso indica que los datos de Cáncer pueden ser filtrados por Tipo Cáncer, mientras que los del cubo Población no. El ejemplo de la medida Tasa es solamente un caso hipotético. No obstante, en GeoCR se cuenta con varias medidas que presentan el mismo inconveniente. Una de ellas es la previamente explicada Tasa de incidencia, mostrada en Código 12.

Considerando la problemática anteriormente descrita, se incluyó una especificación para que GeoMondrian ignore cualquier filtro hecho sobre la dimensión Tipo Cáncer al calcular el valor de la medida Cantidad de habitantes dentro de la fórmula de Tasa de incidencia. En términos prácticos, se aplicó la función MDX Aggregate sobre el elemento All Tipos de Tipo Cáncer para la medida Cantidad de habitantes, como se aprecia en el Código 12. All Tipos es el miembro superior de la jerarquía asociada a Tipo Cáncer, por lo que su valor para la medida Cantidad de habitantes es el resultado de la agregación de todos sus elementos. Por lo tanto, sin importar los filtros especificados en las consultas, Cantidad de habitantes se calcula como su valor total para todos los tipos (a nivel de distrito). Esto hace que, en cualquier caso donde se involucre a la medida Tasa de incidencia, GeoMondrian calcule el valor de Cantidad de habitantes considerando únicamente los filtros en las dimensiones para las cuales Cantidad de habitantes puede agregarse. Por ejemplo, si se pide Tasa de incidencia para Leucemia en el Distrito A, la herramienta obtiene (1) el valor de Cantidad de habitantes en el Distrito A y (2) el valor de Casos de Leucemia en el Distrito A. Una vez que se tienen esos cálculos, es posible obtener el resultado esperado para la medida en cuestión.

CUBOS VIRTUALES

Como se pudo observar en la seción anterior, una medida calculada en el esquema de GeoMondrian se define dentro del cubo que contiene las medidas que utiliza en su fórmula; sin embargo, en caso de que el cálculo involucre medidas de distintos cubos, es necesario utilizar cubos virtuales. Un cubo virtual es aquel compuesto por un subconjunto de medidas y dimensiones de uno o más cubos base; su característica principal reside en la capacidad para crear un ambiente compartido, propiciando así la mezcla de medidas y dimensiones que inicialmente eran independientes unas de otras, dada la separación entre los cubos. Una ventaja adicional de la utilización de cubos virtuales en GeoMondrian es que la herramienta almacena únicamente la definición del cubo en el esquema y no los datos involucrados en el mismo. Por lo tanto, es posible crear diferentes combinaciones y variantes de cubos existentes sin desperdiciar espacio físico de almacenamiento [MSD07].

Dado que en GeoCR se requirió la creación de medidas calculadas con base en medidas incluidas en cubos distintos, se recurrió a la utilización de cubos virtuales. Por ejemplo, la medida calculada Tasa de incidencia utiliza las medidas Cantidad de habitantes y Casos —tomadas de diferentes cubos base— para poder calcular correctamente su valor. La definición de Tasa de incidencia —que fue explicada utilizando el extracto Código 12— se muestra ya incluida dentro de un cubo virtual en el Código 13 (filas 12 a 14).

Código 13. Uso de cubos virtuales en un esquema de GeoMondrian

```
1  <VirtualCube name="cancer_poblacion">
2     <CubeUsages>
3          <CubeUsage cubeName="cancer" ignoreUnrelatedDimensions="true"/>
4          <CubeUsage cubeName="poblacion"/>
5     </CubeUsages>
6     <VirtualCubeDimension cubeName="cancer" name="Tipo Cáncer"/>
7     <VirtualCubeDimension cubeName="cancer" name="Distritos"/>
8     <!--Otras dimensiones necesarias. -->
9     <VirtualCubeMeasure  cubeName="poblacion"  name="[Measures].[Cantidad
      de habitantes]"/>
10    <VirtualCubeMeasure cubeName="cancer" name="[Measures].[Casos]"/>
11    <VirtualCubeMeasure cubeName="cancer" name="[Measures].[Muertes]"/>
12    <CalculatedMember name="Tasa de incidencia " dimension="Measures">
13         <Formula>
               <!--Verificación  de  que  el  valor  para  "Cantidad  de
               habitantes" sea mayor a cero y diferente de nulo-->
               [Measures].[Casos]/(Aggregate({[Tipo  Cáncer].[All  Tipos]},
               [Measures].[Cantidad de habitantes]))*100000
          </Formula>
14    </CalculatedMember>
15    <!--…-->
16 </VirtualCube>
```

Para crear el cubo virtual dentro de un esquema, se deben definir: los cubos base que serán combinados, las dimensiones y medidas (ambos tipos de elementos pueden provenir de otros cubos base) y, en este caso particular, las fórmulas para las medidas calculadas. El esquema mostrado en el extracto Código 13 —cuyo objetivo es posibilitar la creación de la medida calculada Tasa de incidencia— muestra un cubo virtual llamado Cancer_Poblacion, definido mediante la cláusula VirtualCube (filas 1 y 16); el mismo contiene dimensiones (Tipo Cáncer y Distritos) y medidas (Casos y Muertes) del cubo Cáncer (filas 6, 7, 10 y 11), así como la medida Cantidad de habitantes del cubo Población (fila 9).

Anteriormente se indicó que las dimensiones compartidas son aquellas que pueden ser utilizadas desde cualquier cubo base. El objetivo básico de la dimensión compartida es similar al del cubo virtual: compartir. Por ello es que, así como una dimensión puede estar contenida dentro de diferentes cubos base, un cubo base puede ser referenciado desde distintos cubos virtuales. Incluso su especificación es semejante; para hacer referencia a una dimensión se utiliza la claúsula DimensionUsage, mientras que para poder incluir un cubo base dentro de un cubo virtual se especifica CubeUsages. En el extracto Código 13, los cubos base Cáncer y Población se especifican dentro de la claúsula CubeUsages (filas 2 y 5), donde cada cubo se define individualmente mediante la etiqueta CubeUsage (filas 3-4). Las dimensiones y medidas del cubo referenciado pueden ser accedidas al indicar su nombre y cubo por medio de las cláusulas VirtualCubeDimension y VirtualCubeMeasure, respectivamente. Por ejemplo, en Código 13, se utilizan estos identificadores para incluir la dimensión Distritos del cubo Cáncer (fila 7) y la medida Cantidad de habitantes del cubo Población (fila 9).

El esquema diseñado para GeoCR —mostrado en el Anexo H— consta de una estructura jerárquica que involucra cuatro cubos virtuales, cinco cubos base y más de diez dimensiones compartidas. Los nombres de cada uno de estos componentes se exponen con detalle en la Figura 20, donde además puede observarse la jerarquía de tres niveles implícita en el esquema. Por ejemplo, Cancer_Poblacion utiliza la información de los cubos base Cáncer y Población; a su vez, el cubo base Cáncer hace referencia a las dimensiones compartidas Tipo, Sexo, Tiempo, Edad y a las dimensiones espaciales, mientras que el cubo base Población se relaciona con las dimensiones compartidas Sexo, Tiempo, Edad y con las espaciales. El cubo virtual Cancer_Poblacion fue creado para poder calcular las medidas Tasa de incidencia y Tasa de mortalidad, que utilizan medidas de los cubos base Cáncer y Población. El detalle de los cálculos para los que se necesitó crear cada cubo virtual se incluye en la última columna a la derecha de la Figura 20 (llamada Justificación).

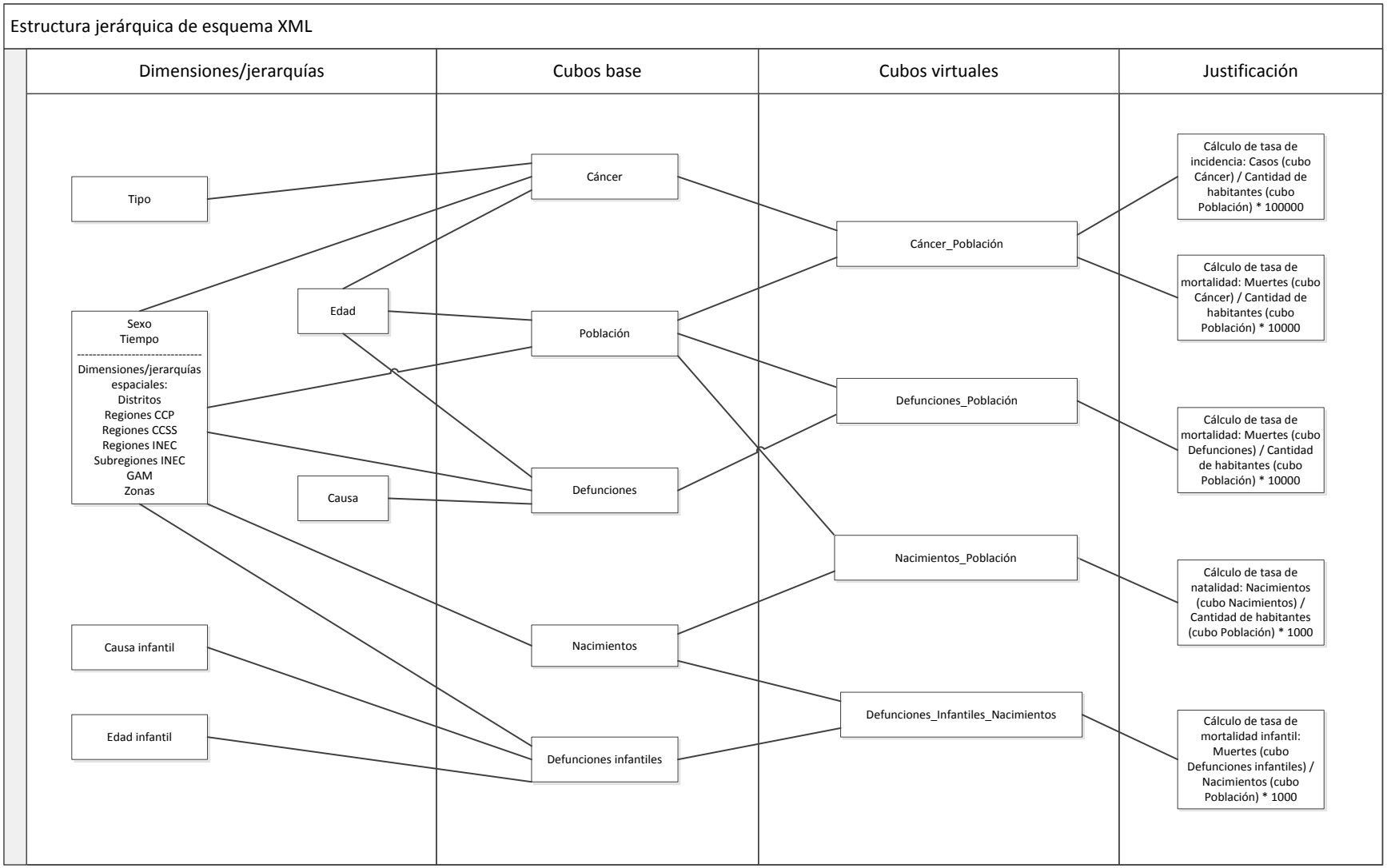


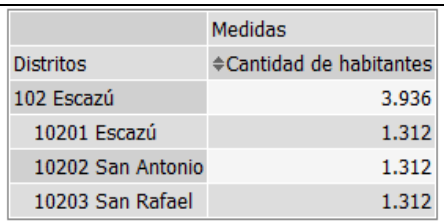
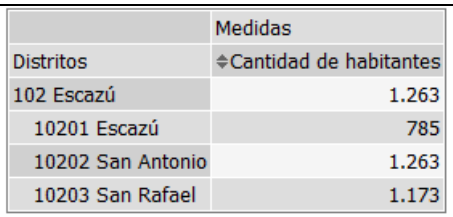
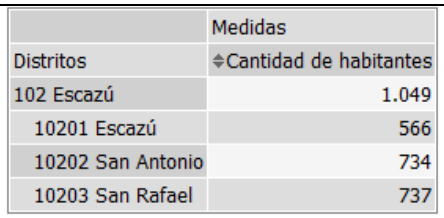
Figura 20. Estructura jerárquica del esquema

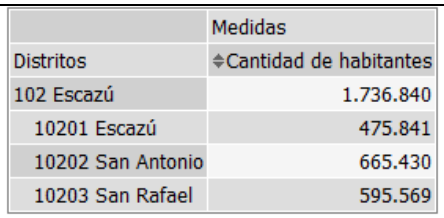
FUNCIONES DE AGREGACIÓN

Cuando se define una medida, GeoMondrian requiere que se especifique una función de agregación; por ejemplo, en el esquema del extracto Código 10 (fila 16) se utiliza la función SUM para indicar que los datos de la medida Muertes del cubo Cáncer deben ser sumados al agregarse. Las medidas Casos y Muertes del cubo Cáncer son aditivas, porque utilizan la suma para agregar los valores en cada una de sus dimensiones. Sin embargo, como se señaló en el marco teórico, existen medidas que producirían resultados semánticamente incorrectos si fuesen procesadas de esta manera. Las medidas semiaditivas son muestra de ello. Por ejemplo, de existir una medida Cantidad de habitantes en el cubo Población, tendría sentido lógico usar la suma sobre ella para agregarla en todas las dimensiones excepto Tiempo. Esta excepción se da porque, por ejemplo, si se tuviera un nivel Quinquenio en Tiempo y se consultara el valor de Cantidad de habitantes para cada elemento de él usando la suma como función de agregación, se obtendrían valores semánticamente incorrectos: el quinquenio 2000-2004 devolvería el valor de los cinco años sumados, cuando lo adecuado sería que mostrara el valor correspondiente al año más reciente del quinquenio seleccionado (2004).

Actualmente, GeoMondrian no incluye mecanismos destinados específicamente al manejo de medidas no aditivas y semiaditivas. No obstante, existen funciones de agregación que pueden ser utilizadas para representar las medidas no aditivas en casos particulares: COUNT, AVG, MIN, MAX y DISTINCT-COUNT. Las funciones de agregación que maneja GeoMondrian, sus efectos y ejemplos asociados se muestran en la Tabla 8.

Tabla 8. Funciones de agregación y ejemplos de resultados obtenidos al aplicarlas

Función de agregación	Ejemplo
COUNT: devuelve la cantidad de sub-elementos.	
AVG: calcula el promedio de los valores correspondientes a los sub-elementos.	
MIN: muestra el menor valor de los sub-elementos.	
MAX: arroja el mayor valor de los sub-elementos.	
DISTINCT-COUNT: presenta la cantidad de sub-elementos sin contar los repetidos.	

SUM: suma los valores de los sub-elementos.	

	Medidas
Distritos	Cantidad de habitantes
102 Escazú	1.736.840
10201 Escazú	475.841
10202 San Antonio	665.430
10203 San Rafael	595.569

	Distritos			
Medidas	102 Escazú	10201 Escazú	10202 San Antonio	10203 San Rafael
Casos	2.041	978	453	610
Muertes	1.096	460	275	361
Cantidad de habitantes	1.736.840	475.841	665.430	595.569
Tasa de mortalidad (por 10000)	6,31	9,667	4,133	6,061
Tasa de incidencia (por 100000)	117,512	205,531	68,076	102,423

Figura 21. Medida no aditiva representada a través de medida calculada

Otra forma de usar medidas no aditivas es a través de medidas calculadas, siempre que puedan ser adaptadas para solventar la necesidad asociada. La Figura 21 muestra los valores de Casos, Muertes, Cantidad de habitantes, Tasa de mortalidad y Tasa de incidencia para el cantón 102 Escazú y sus tres distritos: 10201 Escazú, 10202 San Antonio y 10203 San Rafael. La medida Tasa de mortalidad es el resultado de dividir Muertes entre Cantidad de habitantes y multiplicar por 10000; en el caso de 102 Escazú esto es 6,310 (1096/1736840*10000). Sin embargo, no hay una función de agregación que pueda calcular ese resultado a partir de los valores de Tasa de mortalidad de los miembros del nivel más bajo (Distritos): 9,667 (10201 Escazú), 4,133 (10202 San Antonio) y 6,061 (10203 San Rafael). Tasa de mortalidad es entonces una medida no aditiva a la cual, dadas sus características, es posible representar por medio de una fórmula, pero no a través de una de las funciones de agregación. Por lo tanto, la manera de

presentar los resultados semánticamente correctos es generando su valor para cada celda individualmente, lo cual se logra al definirla como medida calculada.

2. CONSULTAS (GEOMDX)

GeoMondrian emplea el lenguaje MDX para realizar consultas sobre los cubos definidos en el esquema XML. Los resultados de las consultas se cargan desde el almacén de datos hacia los cubos, utilizando la memoria principal como medio de almacenamiento destino. Por lo tanto, en caso de que la ejecución de una consulta se repita, los datos se obtienen directamente de la memoria principal; ahorrándose así el acceso al almacén y los cálculos asociados.

Una consulta en MDX contiene la siguiente información [MSD03]: (1) los miembros de los ejes que van a ser desplegados, (2) el nombre del cubo sobre el cual se hace la consulta y (3) el miembro que se utiliza para delimitar los datos retornados por la consulta (en caso de que se aplique *slice-and-dice*).

Código 14. Consulta simple en MDX

```
1  select {[Measures].[Muertes]} on columns,
2  filter ({[Distritos].Members}, [Measures].[Muertes] > 500) on
   rows
3  from [Cancer]
4  where [Tiempo].[2005]
```

En el extracto Código 14 se muestra una consulta simple en MDX, donde los elementos de la dimensión Distritos y la medida Muertes son los miembros de los ejes a desplegar (fila 2). Cáncer es el cubo que se va a utilizar en la consulta (fila 3) y el elemento 2005 de la dimensión Tiempo (fila 4) es utilizado para filtrar los resultados por ese año específico (*slice-and-dice*). Los alias ON COLUMNS y ON ROWS sirven para especificar los valores que se asignan en los ejes X y Y de la tabla de resultados. Una consulta también puede incluir un filtro para que se desplieguen únicamente los elementos que cumplen con cierta condición. En el caso particular del ejemplo presentado en el Código 14, se incluye un filtro para que se muestren sólo los miembros de Distritos que tienen más de 500 muertes (fila 2).

La posibilidad de ejecutar operaciones sobre datos espaciales constituye una de las principales características de GeoMondrian. Como un medio para plasmarla en el sistema, la herramienta potencia el lenguaje MDX a través del aditamento de funciones dedicadas al análisis geoespacial, razón por la que se le llama lenguaje **GeoMDX**. Las funciones espaciales incluidas aportan alternativas para realizar operaciones sobre las geometrías, como el cálculo de centroides, intersecciones, áreas y distancias.

La anteriormente mencionada aplicación de filtros (fila 2 del Código 14) puede ser utilizada en conjunto con las funciones espaciales incluidas en GeoMondrian. Un ejemplo de ello puede observarse en el Código 15, donde se presentan dos funciones de este tipo para que los resultados de la consulta muestren sólo las provincias cuya área supere los 7500 km^2. Para calcular el área, se necesita un sistema de coordenadas proyectado que utilice el kilómetro como la unidad de medida (código 32617 para el sistema WGS84/UTM zone 17N); sin

embargo, el sistema utilizado en la representación de las geometrías almacenadas en PostGIS (código 4326 para el sistema WGS84) se basa en latitud y longitud. Dado esto, los datos tienen que transformarse de un sistema al otro por medio de la función ST_Transform. Finalmente, el área se calcula en kilómetros cuadrados por medio de la función ST_Area.

Código 15. Consulta con función GeoMDX

```
1  select {[Measures].[Muertes]} on columns,
2  filter ({[Distritos].Children},
   ST_Area(ST_Transform([Distritos].CurrentMember.Properties("geom")
   ,4326,32617))/1E6 > 7500) on rows
3  from [Cancer]
4  where [Tiempo].[2005]
```

Código 16. Consulta MDX que crea una medida adicional

```
1  with member [Measures].[Diferencia casos muertes] as
   '([Measures].[Casos] – [Measures].[Muertes])'
2  select {[Measures].[Casos], [Measures].[Muertes],
   [Measures].[Diferencia casos muertes]} on columns,
3  {[Distritos].Children} on rows
4  from [cancer_poblacion]
5  where [Tiempo].[2005]
```

El lenguaje MDX permite crear medidas calculadas dentro de la misma consulta, lo cual constituye una opción útil para el análisis. En determinado caso, se utiliza el alias WITH MEMBER para nombrar la medida y luego se incluye AS para definir la fórmula asociada, que

puede contener otras medidas definidas dentro del cubo usado. Por ejemplo, el Código 16 muestra la creación de la medida Diferencia casos muertes a partir de una resta que involucra otras dos medidas: Casos y Muertes.

La definición de una medida calculada también puede incluir funciones espaciales. En la consulta mostrada en el Código 17, se crea la medida llamada Área con el objetivo de que muestre el área correspondiente a cada uno de los elementos. Una vez más, las funciones espaciales utilizadas son ST_Transform y ST_Area.

Código 17. Consulta que utiliza una función GeoMDX para crear una medida adicional

```
1  with member [Measures].[Área] as
   'ST_Area(ST_Transform([Distritos].CurrentMember.Properties("geom"
   ),4326,32617))'
2  select {[Measures].[Casos], [Measures].[Muertes],
   [Measures].[Área]} on columns,
3  {[Distritos].Children} on rows
4  from [cancer_poblacion]
5  where [Tiempo].[2005]
```

Las funciones espaciales incluidas en GeoMondrian son el pilar de su análisis espacial. Como se ha visto, pueden ser incluidas en la consulta tanto para filtrar datos como para crear nuevas medidas. En el Anexo B se muestran las funciones espaciales que ofrece GeoMondrian, así como la descripción de sus parámetros y los resultados obtenidos después de su ejecución.

CAPÍTULO VI: INTEGRACIÓN DE CUBOS ESPACIALES CON LA HERRAMIENTA CLIENTE

Para el despliegue de los datos que produce la ejecución de consultas, GeoMondrian utiliza **JPivot** (integrado en Mondrian) [SOU04], un visualizador que presenta los resultados mediante tablas y gráficos. Además, la herramienta proporciona opciones gráficas para crear consultas a través de la especificación de medidas y dimensiones; en dado caso, el código en MDX asociado a la consulta se genera automáticamente y es accesible desde la interfaz a través del editor de MDX, tal y como se muestra en la ventana superior de la Figura 22. Si se modifica la consulta (por ejemplo, al aplicar alguna operación sobre los datos), el código MDX se actualiza automáticamente.

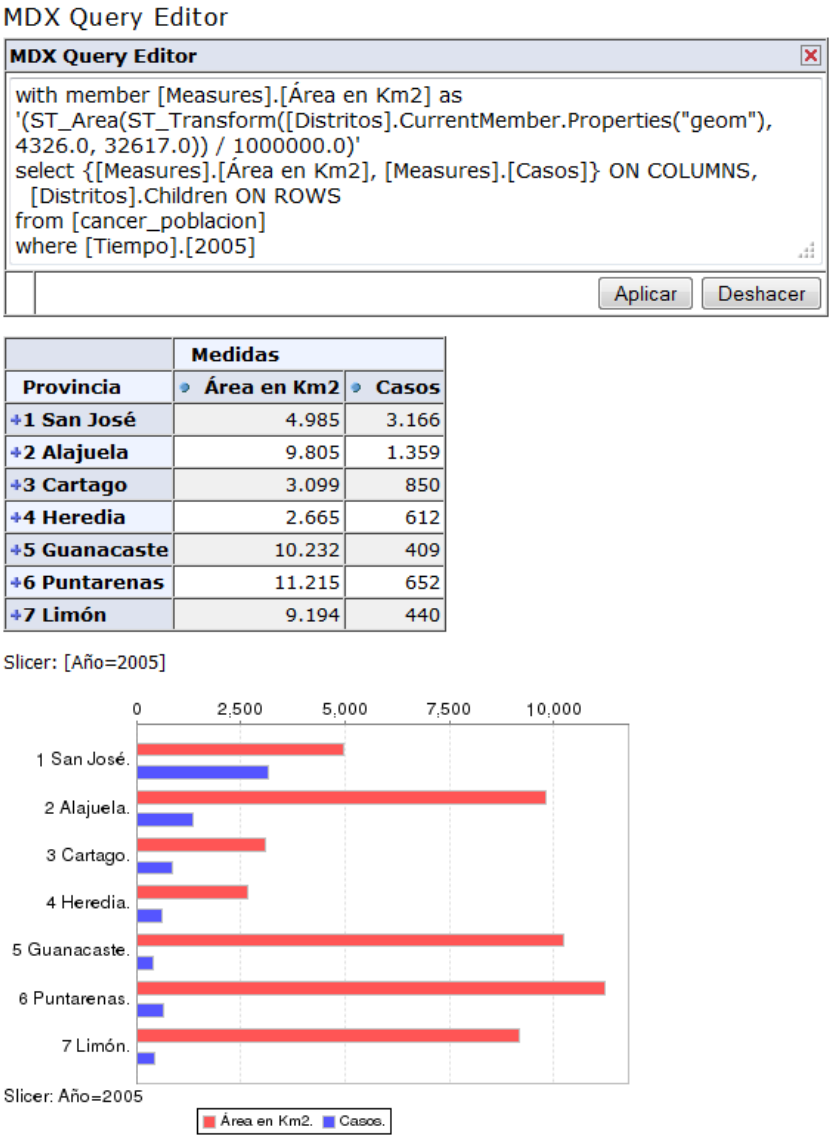


Figura 22. Consulta GeoMDX y su resultado gráfico en JPivot

JPivot permite la ejecución de las operaciones *roll-up*, *drill-down*, *pivot* y *slice-and-dice*. La Figura 23 muestra la forma en que el visualizador despliega los datos al ejecutar las operaciones *roll-up* y *drill-down*. Cuando se aplica *drill-down* sobre el elemento 5 Brunca, se

muestran los miembros del nivel inferior. Por el contrario, esos sub-elementos de 5 Brunca se ocultan cuando se les aplica *roll-up*.

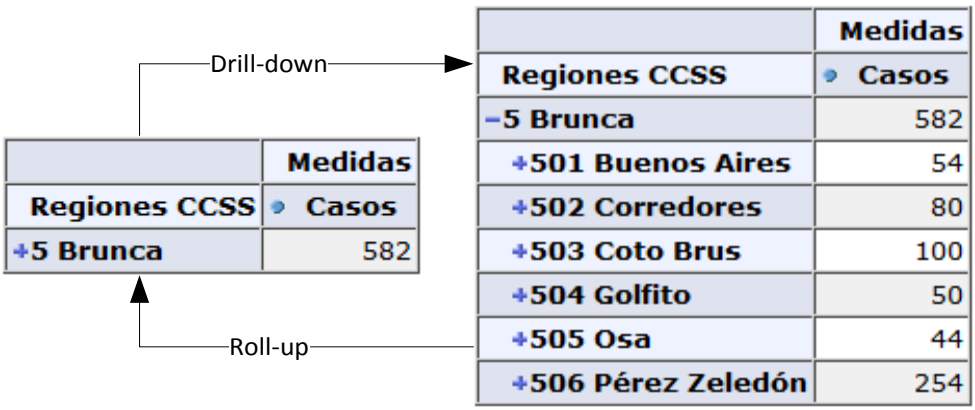


Figura 23. Drill-down y roll-up en JPivot

En SOLAP, el aspecto de visualización es el más significativo y a la vez el más desafiante [BIM07]. Por lo tanto, como requerimiento básico, la herramienta cliente SOLAP debe permitir la navegación entre niveles cartográficos y textuales, así como el soporte del análisis multidimensional mediante la utilización de mapas, tablas y gráficos de forma sincronizada [BIM07, RIV05]. Aunque JPivot tiene la capacidad de mostrar los datos en tablas y gráficos simultáneamente, no cuenta con un despliegue de datos espaciales sobre mapas. Esa carencia —que resulta ser la diferencia entre OLAP y SOLAP— causó que JPivot fuera desestimada como herramienta cliente del proyecto.

La exclusión de JPivot hizo que se llevara a cabo una búsqueda de herramientas cliente SOLAP, donde la mayoría de las encontradas fueron propietarias. Una de ellas, *JMap Spatial OLAP* —posteriormente llamada *Map4Decision*— es pregonada como la primera tecnología web que integró bases de datos espaciales dentro de un ambiente de soporte de decisiones

[BED09]. Otra herramienta, llamada *GeWOlap,* resalta por su arquitectura de tres capas: (1) sistema de gestión de base de datos objeto-relacional, (2) servidor OLAP y (3) capa cliente que combina OLAP y Sistemas de Información Geográfica (GIS - *Geographic Information Systems*), la cual implementa utilizando JPivot [BIM07]. También se han desarrollado herramientas a la medida para un área de análisis específica; por ejemplo, la propuesta por Scotch y Parmanto combina OLAP con GIS a fin de analizar información de la salud pública [SCO05]. Además, equipos de investigación han propuesto alternativas para modelar y consultar datos espaciales [BED09]. Sin embargo, al ser estas herramientas propietarias, no ofrecen acceso libre para su utilización; eso, sumado al requerimiento del proyecto que demanda la utilización de *software* libre, hizo que se descartara el uso de las mismas para la implementación de GeoCR.

Por otra parte, en el ambiente de *software* libre, la cantidad de herramientas disponibles de tipo SOLAP cliente para el análisis y trazado de datos espaciales en mapas es escasa. Dada esta carencia, en un inicio se incorporó en GeoMondrian la opción de crear los mapas usando el sistema R [INS13] con base en los resultados obtenidos a partir de las consultas realizadas. El procedimiento se dividió en dos etapas: primero, se ejecutó la consulta en GeoMondrian con el fin de obtener el conjunto de datos y luego, se guardaron los resultados en archivos XLS temporales para que el sistema R generara los mapas. No obstante, esta implementación fue ineficiente y poco flexible, además de que no permitió realizar las manipulaciones sobre el mapa requieridas por el proyecto. Posteriormente, se analizó la posibilidad de utilizar Google

Fusion Tables[7], sin embargo, su uso demandaba la importación de los datos para cada consulta de interés, lo cual constituye una limitación al implementar medidas calculadas y consultas *ad-hoc*.

Una de las principales opciones libres halladas fue la desarrollada por la compañía *SpagoBI Competency Center*, que ofrece un conjunto de herramientas para inteligencia de negocios [SPA13]. Basada en más de treinta motores analíticos desarrollados por la empresa, *SpagoBI* brinda diferentes opciones para manipular y analizar información. En cuanto al ámbito geográfico, la herramienta ofrece operaciones propias de GIS, como el cálculo de áreas y distancias, y permite representar medidas en las divisiones territoriales mostradas en el espacio geográfico mediante un mapa de calor (*heatmap*) basado en una escala de colores. Sin embargo, aunque *SpagoBI* tiene la capacidad para presentar datos agregados mediante la selección de diferentes capas, no contiene funcionalidad para moverse por los niveles mediante acciones directas sobre el mapa. Además, cuenta con la limitación de que sus componentes de visualización espacial y operaciones OLAP están separados.

Tras finalizar la búsqueda de opciones existentes y valorar soluciones como las mencionadas, se seleccionó la solución **GeoOLAP**[8]. Más allá de que esta herramienta —desarrollada

[7] Google, "Fusion Tables". [En línea]. Disponible: http://www.google.com/drive/apps.html#fusiontables. [Último acceso: 2 de noviembre, 2013]
[8] P. Mauduit, "GeoBI". GitHub. [En línea]. Disponible: https://github.com/pmauduit/GeoBI. [Último acceso: 30 de abril, 2013]

originalmente por la organización Camptocamp[9]— ya no continúa siendo impulsada por sus autores, su elección se fundamentó en las características que posee para la ejecución de algunas las operaciones básicas OLAP que, en conjunto con la visualización de mapas, permiten representar los datos de forma gráfica e incorporarlos en el análisis. Cabe destacar que la definición de los cubos —explicada en el capítulo anterior— es indispensable para el funcionamiento de esta herramienta. Las instrucciones para la instalación de GeoOLAP se muestran en el Anexo G.

1. IMPLEMENTACIÓN CON GEOOLAP

La principal característica de GeoOLAP es su capacidad para manejar y representar dimensiones espaciales, lo cual realiza a través de la extracción de datos correspondientes a coordenadas y la presentación de estos sobre un mapa. La herramienta también permite combinar dimensiones espaciales y convencionales en una misma consulta, filtrando los datos según se le indique. Además, despliega los datos de tres maneras distintas en forma simultánea: gráficos, mapas y tablas. Un ejemplo se puede observar en la Figura 24, que muestra los resultados generados en GeoOLAP tras una consulta sobre la cantidad de casos de cáncer distribuida por sexo en diferentes provincias.

[9] Y. Jacolin y A. Gioia, "GeoBI Initiative: The open source location intelligence ecosystem" (9 de noviembre, 2010). [En línea]. Disponible: http://www.spagoworld.org/spw-resources/Resources/Presentations/GeoBI@fOSSa2010.pdf. [Último acceso: 30 de abril, 2013]

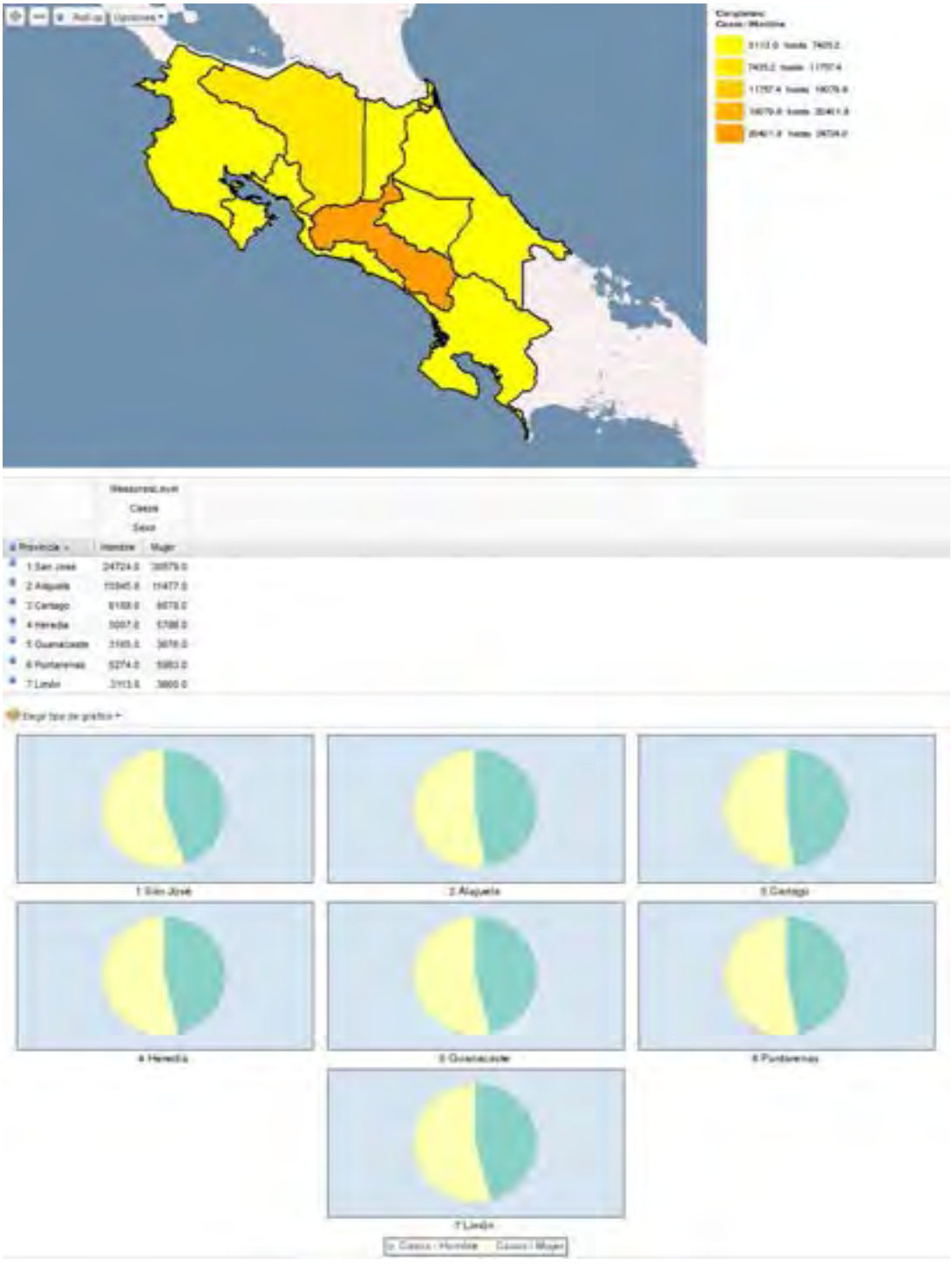


Figura 24. Despliegue de resultados en GeoOLAP

A pesar de que sus funciones están centradas en el uso del mapa, GeoOLAP también ofrece el despliegue de los datos a través de tablas y gráficos. La tabla que se puede ver en la Figura 24 muestra los datos en su versión convencional, algo similar al estilo de la que utiliza GeoMondrian, pero más limitada en cuanto a opciones; por ejemplo, no cuenta con la operación *pivot*. No obstante, es posible aplicar acciones en la tabla para afectar el mapa, porque cuenta con funciones que hacen posible ejecutar *drill-down* y *roll-up* sobre sus miembros. En cuanto a los gráficos, GeoOLAP permite seleccionar uno entre cuatro tipos disponibles: circular por columna, circular por fila (el mostrado en la Figura 24), barras horizontales y barras verticales. Tanto la tabla como los gráficos se muestran en secciones separadas bajo el área del mapa, lo cual puede ser visto en la Figura 24. Las operaciones ejecutadas sobre el mapa se traducen en cambios automáticos en estas dos representaciones, que están siempre presentes.

Por otra parte, el despliegue de los elementos espaciales sobre el mapa puede ser apreciado en la sección superior de la Figura 24. La escala de colores utilizada se crea dinámicamente con cada consulta, tomando como base los valores de la medida consultada. Esto significa que se crean clases —equivalentes a un rango de valores— a las cuales se les asigna un color determinado, de manera que su escala corresponda con la incidencia de la medida seleccionada. Los elementos pertenecientes a las clases generadas son llamados **coropletas** y su simbología se muestra a la derecha del mapa. En la Figura 25 se presenta el detalle de las clases de las coropletas, donde cada una es representada por un color distinto.

Figura 25. Coropletas en GeoOLAP

GeoOLAP proporciona la opción para descargar los datos en formato PDF. El archivo generado contiene el mapa, los gráficos y la tabla. De esta manera, los usuarios podrán almacenar una copia local de los resultados de su consulta. Además, se les ofrece la opción de agregar el título que será incluido al inicio del archivo, así como algunos comentarios adicionales.

OPERACIONES

Entre las principales operaciones de GeoOLAP están las siguientes:

- ***Drill-down* y *roll-up*.** *Drill-down* se realiza al pulsar sobre una región del mapa, mientras que *roll-up* se ejecuta al oprimir el botón correspondiente en la parte superior. Por ejemplo, si se despliegan los cantones de `1 San José` sobre el mapa (figura 26a), la acción de *roll-up* muestra sólo esa provincia; es decir, sube un nivel (figura 26b). *Drill-down* produce el efecto contrario.

Figura 26. Roll-up en GeoOLAP

- ***Slice-and-dice.*** Para aplicar la operación *slice-and-dice* se deben elegir miembros específicos de una dimensión (Figura 27a). En el caso de la Figura 27b, se muestra el resultado producido en el mapa luego de seleccionar sólo los datos correspondientes a 3 Cartago y 4 Heredia.

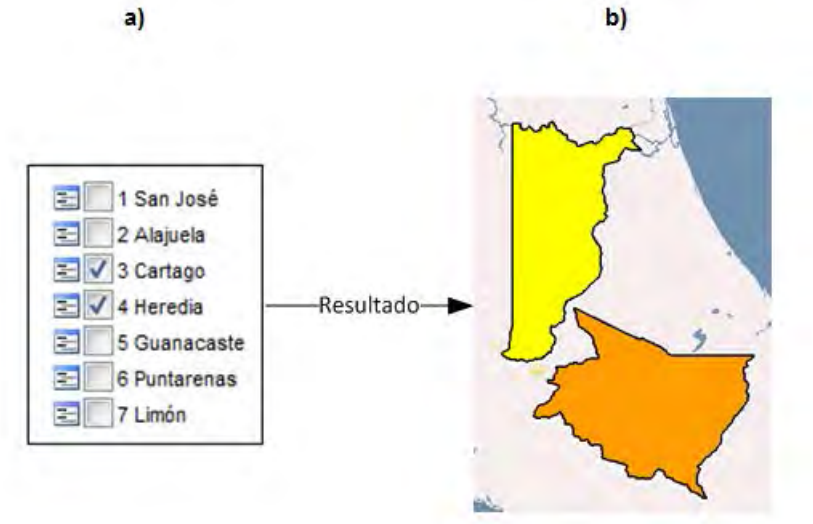


Figura 27. Slice-and-dice en GeoOLAP

- **Combinación de dimensiones espaciales.** No es una operación ampliamente conocida dentro de SOLAP, pero es útil para poder combinar las geometrías correspondientes a distintas dimensiones y desplegarlas en la forma de intersección de dos mapas. GeoOLAP incluye esta operación al permitir la creación de nuevos elementos mediante la combinación de las dimensiones espaciales. Por ejemplo, la Figura 28 muestra el mapa producido al seleccionar la medida Casos del cubo Cáncer para el miembro 6 Puntarenas del nivel Provincias y dos miembros de Regiones INEC: 4 Pacífico Central y 5 Brunca. La ejecución de la consulta crea dos nuevos elementos para su despliegue: (1) 6 Puntarenas 4 Pacífico Central (color naranja en el mapa) y (2) 6 Puntarenas 5 Brunca (color amarillo en el mapa).

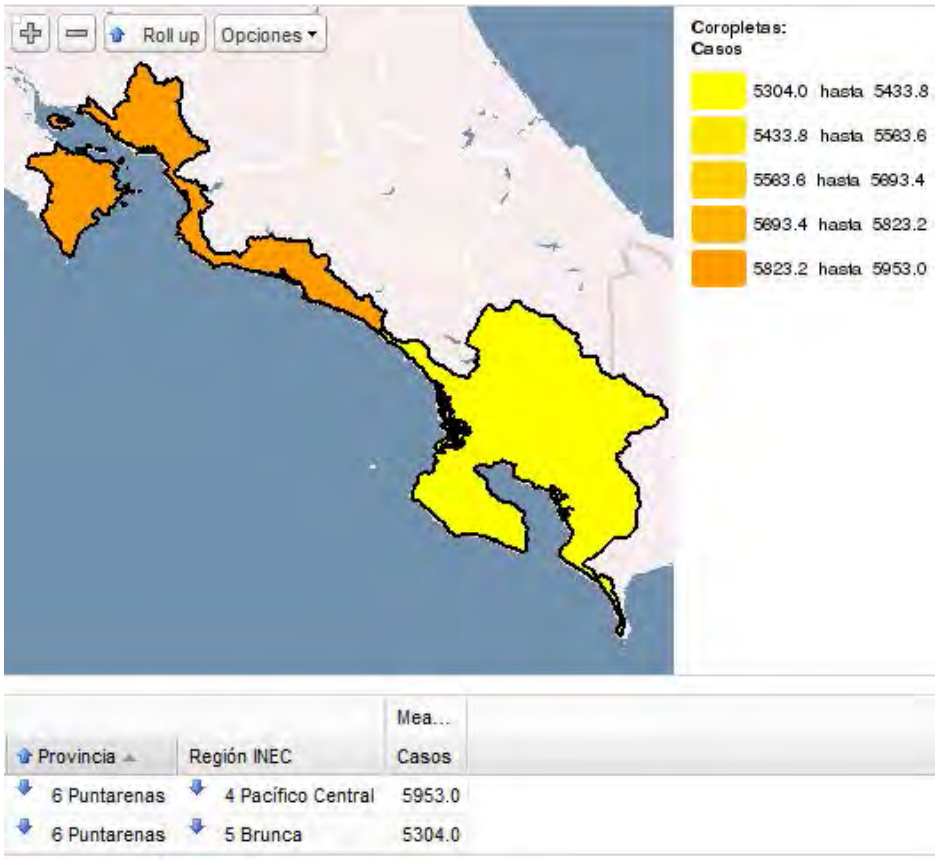


Figura 28. Mapa que combina dimensiones espaciales en GeoOLAP

ASPECTOS DE ALMACENAMIENTO

GeoOLAP —por medio de GeoMondrian— maneja un caché especial para almacenar los componentes del cubo definidos en el esquema. Eso significa que realiza la carga de las medidas y dimensiones —incluyendo a sus miembros— solamente la primera vez que un usuario ingresa a la pantalla de análisis de un enfoque. Entonces, en principio, el sistema guarda la información básica del esquema y la mantiene para evitar que se cargue cada vez que se realice una consulta. Además, los resultados de las consultas también se mantienen en memoria con el fin de evitar el cálculo de los resultados repetidamente y, en lugar de ello, se

extraen directamente del caché. Esto puesto que, como lo indica la documentación de Mondrian sobre su desempeño[10], el tiempo consumido por la herramienta en el cálculo de expresiones usualmente es insignificante si se le compara con el tiempo que tarda ejecutando SQL. La lentitud de la carga inicial se justifica, entonces, por las consultas SQL que la componen, donde el motor OLAP no es eficiente.

Dentro del código de GeoOLAP, existen funciones para controlar el caché de resultados. Tras experimentar con distintos valores, se optó por un espacio con la capacidad de almacenar hasta 7000 elementos y que, en caso de que se supere ese límite, permita que los datos sean guardados en disco. Por su parte, los elementos en el caché se definieron como "eternos", lo cual significa que nunca serán borrados, a menos de que se reinicie el sistema. Esto implica un beneficio en velocidad, porque cada consulta realizada almacena sus resultados en memoria, evitando así que tengan que ser recalculados cada vez que esa misma consulta se ejecuta. No obstante, en caso de que los datos sufran modificaciones constantemente, esto podría convertirse en un factor negativo, ya que los resultados extraídos de la memoria podrían estar obsoletos.

Al final, la decisión sobre el tipo de almacenamiento de resultados —temporal o permanente— tiene que ver con la característica del conjunto de datos; si es actualizado constantemente, se debe contar con tiempos de expiración, de manera que los elementos en memoria sean eliminados cada cierto tiempo y eso obligue al sistema a calcularlos de nuevo,

[10] S. Wood y J. Hyde. Pentaho Mondrian Project. "Pentaho Mondrian Documentation: Optimizing Mondrian Performance", http://mondrian.pentaho.com (noviembre, 2007) [En línea]. Disponible: http://mondrian.pentaho.com/documentation/performance.php [Último acceso: 10 de febrero, 2014].

brindando así resultados actualizados. Empero, dado que los datos utilizados en GeoCR son actualizados pocas veces al año, se optó por definir a los resultados como "eternos" y admitir que se acumulen en el caché. Por lo tanto, el sistema SOLAP es basado en memoria.

ESTILOS PARA LA VISUALIZACIÓN DE LOS RESULTADOS

GeoOLAP ofrece la posibilidad de cambiar varias opciones relacionadas con el despliegue de las coropletas y otros símbolos adicionales por medio de dos paneles específicos en la interfaz. El panel Coropletas permite seleccionar el método de clasificación de las coropletas (intervalos iguales, valores únicos o cuantilos), su rango de colores y la cantidad de clases, mientras que el panel Símbolos adicionales permite desplegar distintos tipos de símbolos —proporcionales, gráficos de barras o gráficos circulares— sobre el mapa, así como cambiar el tamaño del símbolo y su indicador asociado.

En el ejemplo de la Figura 29, se despliegan tanto las coropletas como los símbolos adicionales. Debido a que estos últimos están activados, se muestra el gráfico circular correspondiente a la distribución de casos de cáncer por sexo en cada provincia. La simbología se muestra a la derecha, debajo de las coropletas. Es importante señalar que los símbolos desplegados sobre el mapa no tienen relación directa con los gráficos que se presentan bajo él. Son completamente independientes, lo cual significa que pueden estarse desplegando gráficos de barras en la sección exclusiva de gráficos y al mismo tiempo gráficos circulares como símbolos adicionales sobre el mapa.

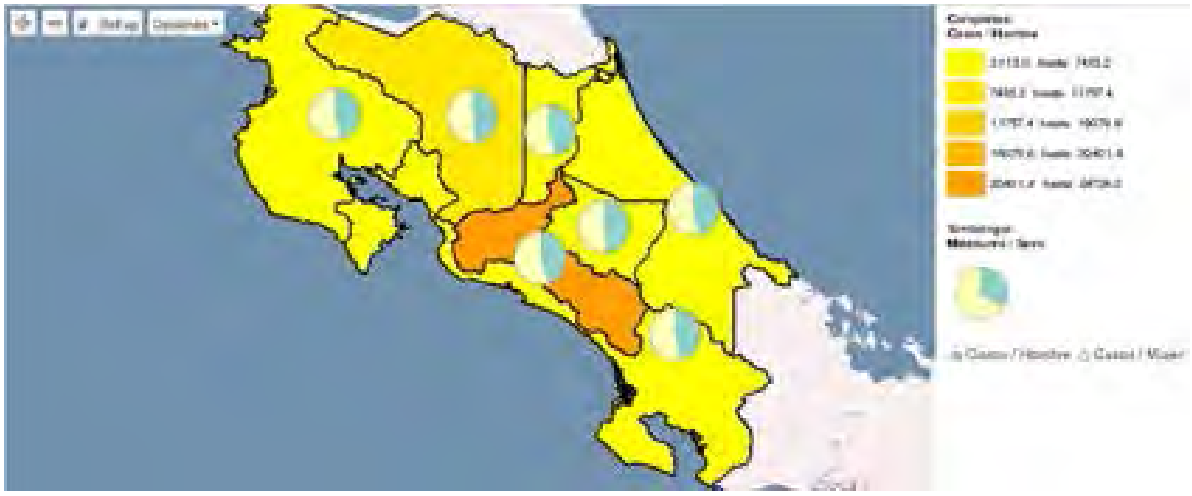


Figura 29. Mapa con gráficos circulares y coropletas en GeoOLAP

MEJORAS INCORPORADAS

Para brindarle al usuario una herramienta fácil de utilizar, se modificaron algunas características de la interfaz de la versión original de GeoOLAP. La versión final utilizada en el proyecto está escrita completamente en español, incluyendo el manejo de tildes y demás caracteres especiales. Asimismo, se ocultó la selección de opciones a las que no se les encontró funcionalidad asociada, como la selección de valores absolutos o relativos de las medidas, en los símbolos adicionales, en las tablas y en los gráficos. También se agregaron barras de desplazamiento en los cuadros de selección de dimensiones, y se añadieron *tooltips* para ciertos botones. Cada uno de los cambios requirió la edición del código fuente y la compilación de la herramienta completa. El Anexo C presenta con detalle las modificaciones realizadas.

Algunos otros cambios implementados en la interfaz tuvieron que ver con ciertas características que podían ser mejoradas. En la versión original, la imagen que contiene la

simbología de las coropletas estaba incompleta cuando se necesitaba desplegar una cantidad determinada de colores asociados. Para resolver ese inconveniente, sus parámetros de tamaño fueron modificados de modo que variaran de acuerdo a la cantidad de coropletas, asegurando así que se mostraran todas. Además, la opción de descargar un archivo en formato PDF tuvo que ser editada, pues, al encontrarse ligada directamente con el enlace de la página, estaba hecha para que funcionara solamente con un único enfoque. Como GeoCR contiene varios cubos, se incorporó una funcionalidad para adaptarla a cualquiera de ellos; con ese cambio, se logró que las impresiones en PDF pudieran llevarse a cabo indistintamente del enfoque de análisis en que se encuentre el usuario.

2. INCORPORACIÓN AL SITIO WEB

El sistema desarrollado incorpora la interacción entre componentes de varios cubos. Como el contexto de visualización ligado a cada uno es distinto, se requirió la elaboración de una página especial en el sitio web del CCP[11]. A través de esta —mostrada en la Figura 30— se explica la función del sistema y se presentan enlaces para ingresar a los enfoques de observación ofrecidos: análisis de cáncer, defunciones, nacimientos y mortalidad infantil.

[11] Centro Centroamericano de Población, "Tasas demográficas". [En línea]. Disponible: http://ccp.ucr.ac.cr/tasas_demograficas/tasas.html [Último acceso: 2 de febrero, 2014]

Figura 30. Vista de la página de GeoCR en el sitio web del CCP

Tabla 9. Detalles de las áreas de análisis que ofrece el sitio web del CCP

Área de análisis	**Cubo virtual**	**Medidas**	**Dimensiones temáticas**	**Dimensiones/ jerarquías espaciales**
Cáncer	Cancer_ Poblacion	Cantidad de habitantes Casos Muertes Tasa de incidencia Tasa de mortalidad	Edad Tiempo Sexo Tipo	Distritos Regiones CCP Regiones CCSS Regiones INEC Subregiones INEC GAM Zonas
Defunciones	Defunciones_ Poblacion	Cantidad de habitantes Muertes Tasa de mortalidad	Edad Tiempo Sexo Causa	
Nacimientos	Nacimientos_ Poblacion	Nacimientos Cantidad de habitantes Muertes Tasa de natalidad Tasa de mortalidad infantil	Tiempo Sexo	
Defunciones infantiles	Defunciones_ Infantiles_ Nacimientos	Cantidad de habitantes Muertes Tasa de mortalidad infantil	Causa infantil Edad infantil Tiempo Sexo	

Es importante tener presente que cada área de análisis corresponde a un cubo virtual del esquema de GeoMondrian. Por ejemplo, el enfoque Cáncer proporciona acceso al cubo virtual Cancer_Poblacion, que contiene las medidas Cantidad de habitantes, Casos, Muertes, Tasa de incidencia y Tasa de mortalidad. La Tabla 9 muestra estas características para cada uno de los enfoques ofrecidos, así como las dimensiones temáticas y espaciales.

Junto a la opción de seleccionar uno de los cuatro enfoques, la página incluye un video tutorial y un manual de la herramienta (Figura 30). El video está en línea y puede ser visto directamente en el sitio, mientras que el documento del manual puede ser descargado desde el enlace proporcionado ahí mismo. Dado que el manejo de GeoOLAP puede ser difícil para un usuario no familiarizado con los conceptos de OLAP y datos espaciales, ambos instrumentos pretenden facilitar el aprendizaje del usuario sobre el tema, explicándole el funcionamiento de la herramienta a partir de ejemplos.

CAPÍTULO VII: ESCENARIOS DE ANÁLISIS Y PROBLEMAS ENCONTRADOS

1. ESCENARIOS DE ANÁLISIS

Los siguientes escenarios muestran las funcionalidades alcanzables con JPivot y GeoOLAP, usando GeoMondrian como motor SOLAP para contextos de consulta como los requeridos por el CCP.

ESCENARIO 1 – VISUALIZACIÓN DE RESULTADOS EN DISTINTOS FORMATOS

JPivot y GeoOLAP permiten la visualización de datos mediante tablas y gráficos para cualquier consulta generada. Además, GeoOLAP ofrece la posibilidad de ilustrar los resultados en mapas. Tomando como base una consulta de ejemplo para obtener la cantidad de casos de cáncer reportados para todos los tipos y géneros durante el 2005 en cada una de las provincias, en JPivot y GeoOLAP es posible seleccionar los siguientes elementos:

- Dimensiones:
    - Distritos (todos los miembros del nivel Provincias).
    - Tiempo (2005).
    - Sexo (todos los miembros).
- Medida:
    - Casos

Resultados en JPivot:

Al ejecutar la consulta, JPivot muestra de forma simultánea la tabla y el gráfico con los valores resultantes, tal y como se muestra en la Figura 31. En ambas representaciones se tienen los casos de hombres y mujeres por separado, así como el total uniendo ambos géneros.

	Medidas		
	Casos		
	Sexo		
Distritos	-Ambos sexos	Hombre	Mujer
-Todas las provincias	7.488	3.455	4.033
+1 San José	3.166	1.427	1.739
+2 Alajuela	1.359	652	707
+3 Cartago	850	415	435
+4 Heredia	612	269	343
+5 Guanacaste	409	193	216
+6 Puntarenas	652	298	354
+7 Limón	440	201	239

Slicer: [Año=2005]

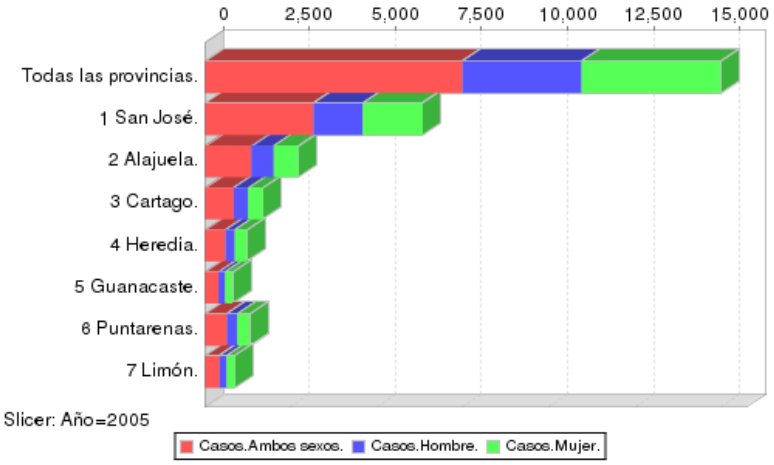


Figura 31. Resultado de escenario 1 en JPivot

Resultados en GeoOLAP:

Por su parte, GeoOLAP muestra mediante tablas, gráficos y mapas únicamente el total de casos para hombres y mujeres por separado, como se observa en la Figura 32. En ella se muestran los casos de hombres mediante las coropletas, mientras que los casos de las mujeres se visualizan con símbolos proporcionales.

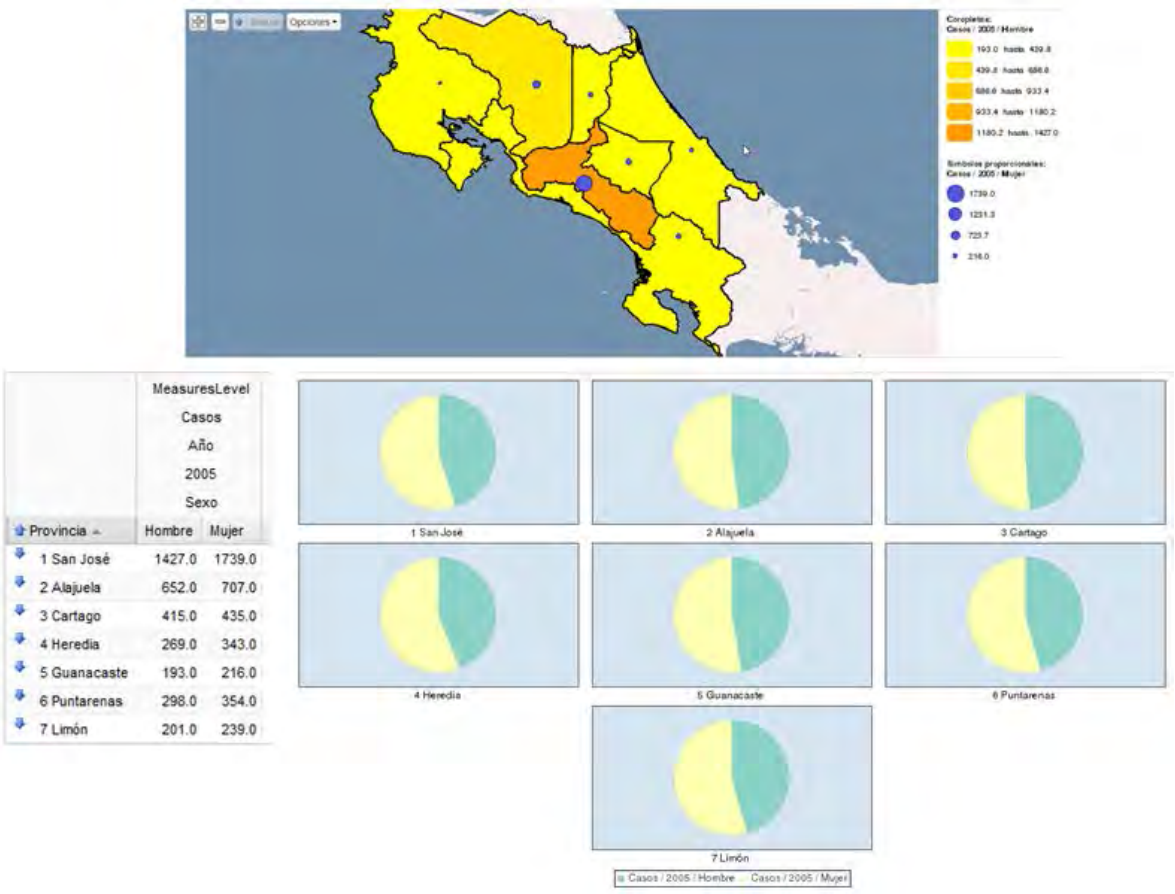


Figura 32. Resultado de escenario 1 en GeoOLAP

ESCENARIO 2 – OPERACIONES BÁSICAS: DRILL-DOWN

Una de las funcionalidades básicas que debe ofrecer una herramienta OLAP es poder realizar *drill-down* sobre las jerarquías de las diferentes dimensiones involucradas. Por ejemplo, utilizando la misma consulta y resultados del Escenario 1, es posible invocar la operación *drill-down* sobre la provincia Cartago.

Resultados en JPivot:

La Figura 33 muestra el resultado obtenido luego de aplicar *drill-down* sobre Cartago. Cabe resaltar que JPivot continúa mostrando los otros miembros del nivel Provincia a los cuales no se les aplicó *drill-down*, permitiendo a su vez aplicar la misma función sobre esos otros miembros en forma simultánea.

	Medidas		
	Casos		
	Sexo		
Distritos	-Ambos sexos	Hombre	Mujer
-Todas las provincias	7.488	3.455	4.033
+1 San José	3.166	1.427	1.739
+2 Alajuela	1.359	652	707
-3 Cartago	850	415	435
+301 Cartago	324	155	169
+302 Paraíso	77	37	40
+303 La Unión	139	72	67
+304 Jiménez	26	11	15
+305 Turrialba	140	68	72
+306 Alvarado	18	13	5
+307 Oreamuno	62	28	34
+308 El Guarco	64	31	33
+4 Heredia	612	269	343
+5 Guanacaste	409	193	216
+6 Puntarenas	652	298	354
+7 Limón	440	201	239

Slicer: [Año=2005]

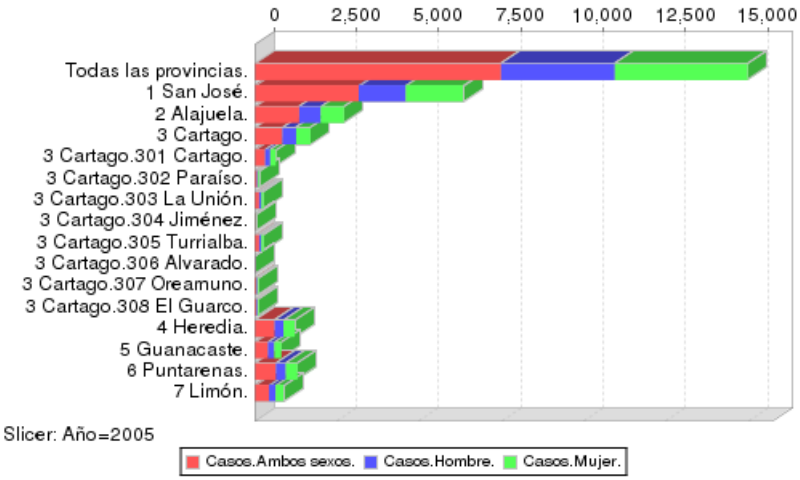


Figura 33. Resultado de escenario 2 en JPivot

Resultados en GeoOLAP:

La Figura 34 ilustra como GeoOLAP, a diferencia de JPivot, descarta los otros miembros del nivel Provincia, aislando únicamente a 3 Cartago en el resultado final del *drill-down*.

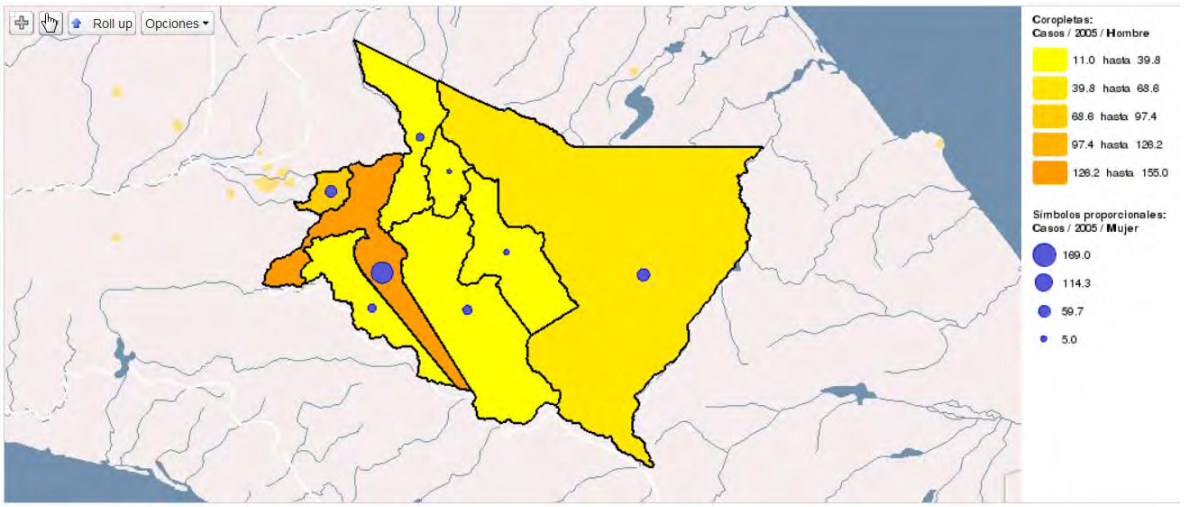


Figura 34. Resultado de escenario 2 en GeoOLAP

ESCENARIO 3 – OPERACIONES BÁSICAS: ROLL-UP

Al igual que *drill-down*, *roll-up* es una operación básica en OLAP. Utilizando la misma consulta del Escenario 1, pero tomando como base los resultados del Escenario 2, es posible invocar *roll-up* sobre la provincia Cartago. En JPivot, *roll-up* se invoca pulsando el símbolo a la izquierda de Cartago, mientras que en GeoOLAP se puede ejecutar presionar el botón Roll up del mapa (o en la flecha a la izquierda de Cantón en la tabla).

Resultados en JPivot:

La Figura 35 muestra que, luego de aplicar *roll-up*, se suman todos los valores de los cantones de la provincia de Cartago para ambos géneros (hombres y mujeres). Estos resultados son los mismos que se muestran en la Figura 31 para 3 Cartago. Al igual que para *drill-down*, JPivot mantiene el estado de los otros miembros del nivel provincia.

	Medidas		
	Casos		
	Sexo		
Distritos	-Ambos sexos	Hombre	Mujer
-Todas las provincias	7.488	3.455	4.033
+1 San José	3.166	1.427	1.739
+2 Alajuela	1.359	652	707
+3 Cartago	850	415	435
+4 Heredia	612	269	343
+5 Guanacaste	409	193	216
+6 Puntarenas	652	298	354
+7 Limón	440	201	239

Slicer: [Año=2005]

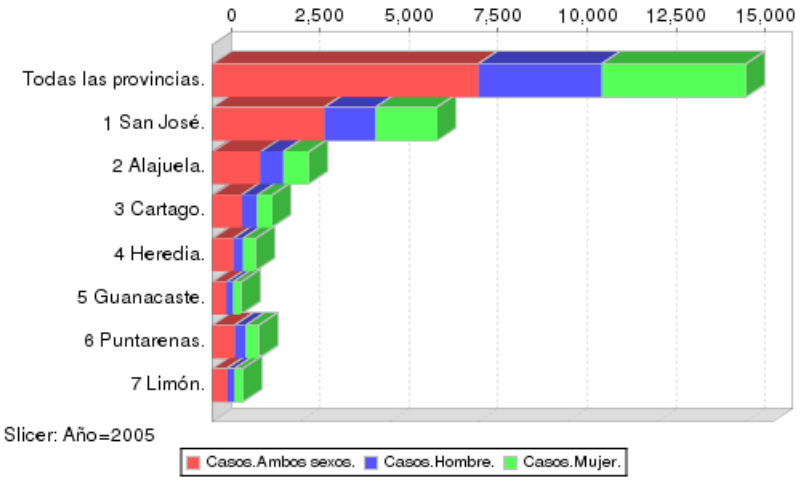


Figura 35. Resultado de escenario 3 en JPivot

Resultados en GeoOLAP:

La Figura 36 muestra que, luego de aplicar *roll-up*, se suman todos los valores de los cantones para ambos géneros —por separado— de la provincia de Cartago. Los resultados son los mismos mostrados en la Figura 32 para 3 Cartago. Al igual que en *drill-down*, se omiten las otras provincias, al enfocarse la función únicamente en el miembro sobre el cual se aplicó.

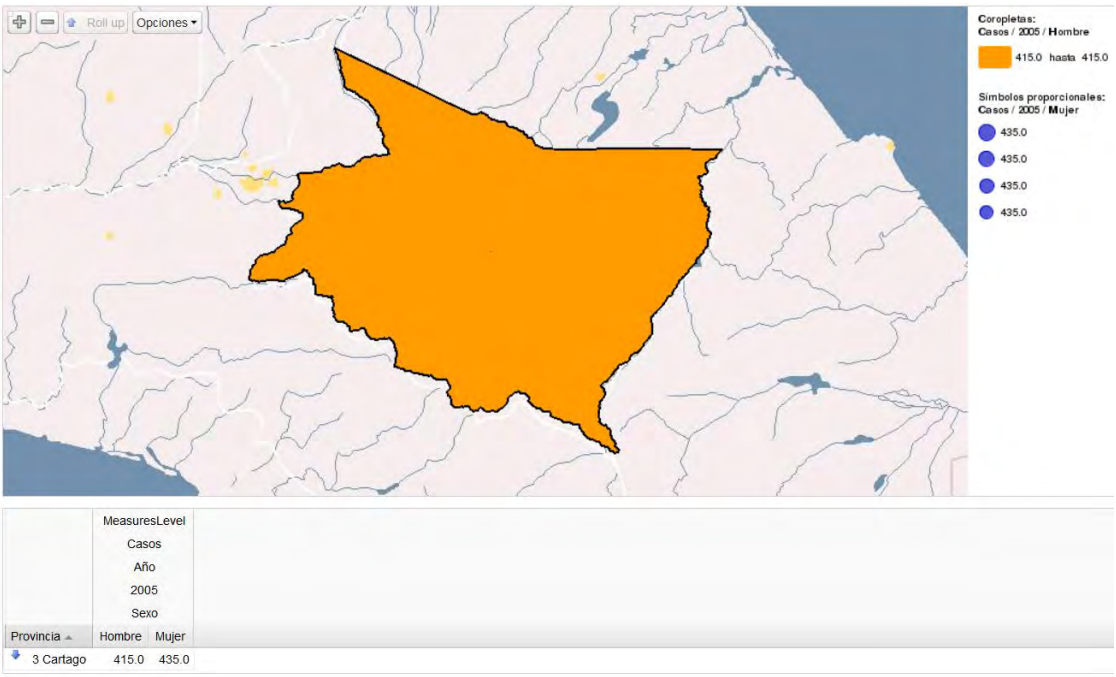


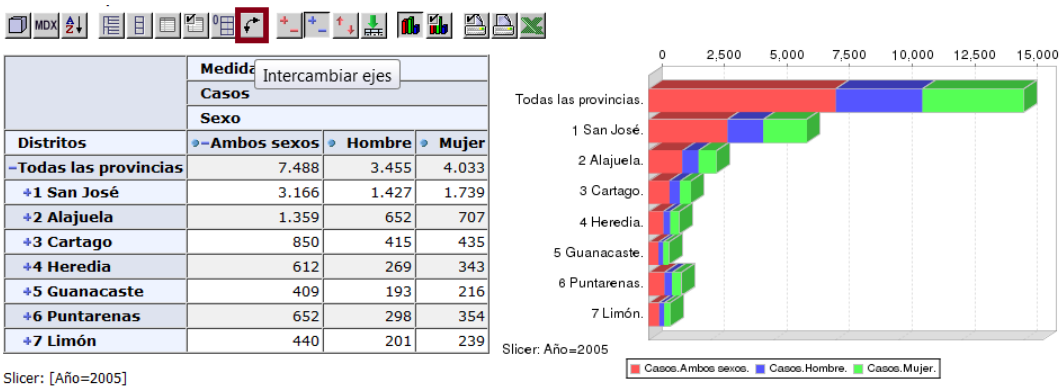
Figura 36. Resultado de escenario 3 en GeoOLAP

ESCENARIO 4 – OPERACIONES BÁSICAS: PIVOT

Otra operación básica de OLAP es *pivot*. Utilizando la misma consulta y resultados del Escenario 1, es posible aplicar *pivot* para rotar los ejes. En JPivot, esta operación se ejecuta mediante el botón Intercambiar ejes; mientras que en GeoOLAP, únicamente es posible aplicarla sobre los gráficos, intercambiando entre Circular por fila o Circular por columna.

Resultados en JPivot:

La Figura 37 muestra que, luego de utilizar la opción Intercambiar ejes de JPivot, se invierten los ejes de la tabla y el gráfico. No se muestran los casos de cáncer por provincia para hombres y mujeres, sino que se observan para cada género los casos por provincia. Los gráficos también se invierten y ya no se dibujan ocho columnas divididas en tres categorías, sino que se muestran tres columnas divididas en ocho categorías.



		Distritos							
Medidas	Sexo	-Todas las provincias	+1 San José	+2 Alajuela	+3 Cartago	+4 Heredia	+5 Guanacaste	+6 Puntarenas	+7 Limón
Casos	-Ambos sexos	7.488	3.166	1.359	850	612	409	652	440
	Hombre	3.455	1.427	652	415	269	193	298	201
	Mujer	4.033	1.739	707	435	343	216	354	239

Slicer: [Año=2005]

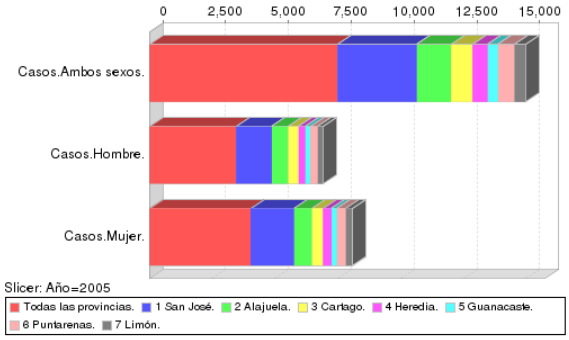


Figura 37. Resultado de escenario 4 en JPivot

Resultados en GeoOLAP:

La Figura 38 muestra la forma en que varían los resultados en GeoOLAP al seleccionar el despliegue de los gráficos por filas o por columnas. A diferencia de JPivot, GeoOLAP no sincroniza la tabla con el intercambio de ejes, de modo que sólo es posible aplicarle *pivot* a los gráficos.

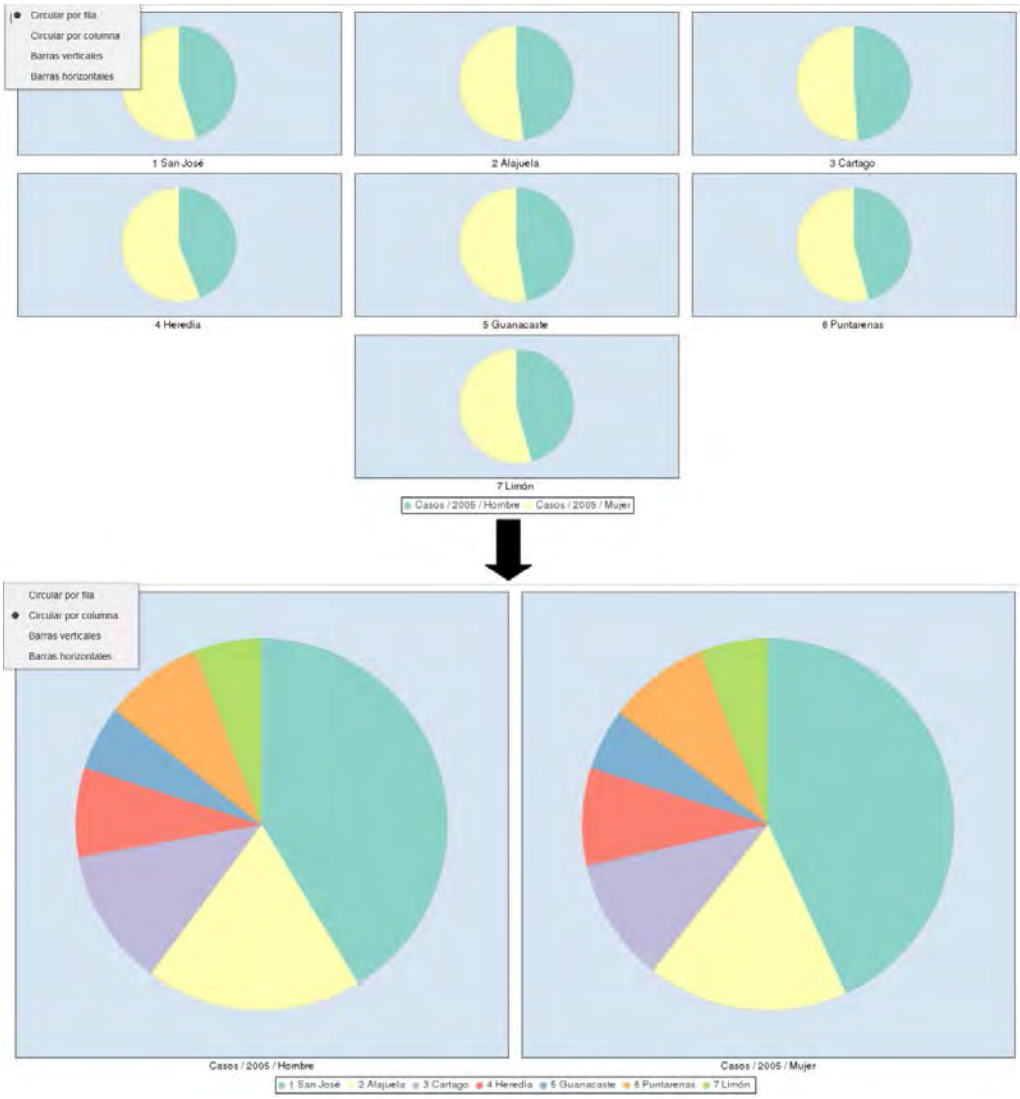


Figura 38. Resultado de escenario 4 en GeoOLAP

ESCENARIO 5 – USO SIMULTÁNEO DE DIMENSIONES ESPACIALES

JPivot y GeoOLAP permiten incluir en una misma consulta dos o más dimensiones espaciales con el objetivo de poder visualizar la intersección de miembros pertenecientes a clasificaciones de áreas geográficas que no tienen relación directa entre sí. Por ejemplo, tomando como base una consulta para obtener las defunciones durante el año 2009 en cada una de las regiones del INEC y provincias a las cuales corresponden, en JPivot y GeoOLAP es posible seleccionar los siguientes elementos:

- Dimensiones:
    - Distritos (todos los miembros del nivel Provincias).
    - Regiones INEC (todos los miembros)
    - Tiempo (2009).
- Medida:
    - Muertes.

Resultados en JPivot:

La Figura 39 muestra el efecto obtenido al combinar dos dimensiones espaciales en una misma consulta. En la tabla y en los gráficos, la segunda dimensión espacial seleccionada se toma como subconjunto de la primera. Por lo tanto, si alguna región del INEC forma parte de varias provincias, como es el caso de 2 Resto Central, los resultados de esta se dividen en cada una de las provincias a las cuales pertenece.

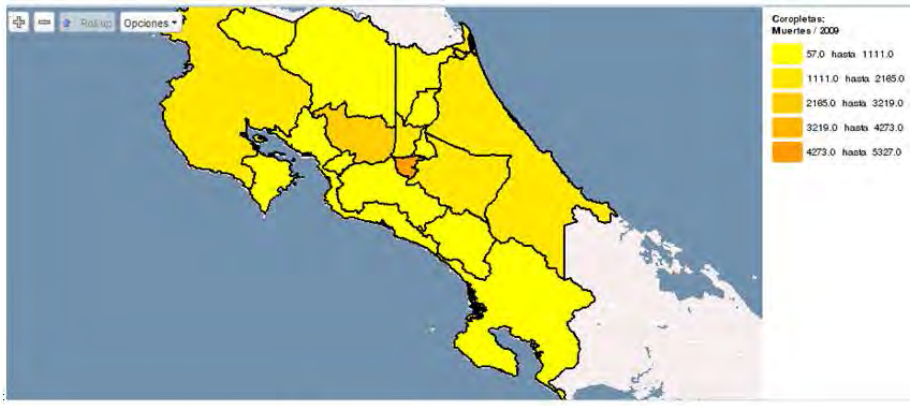
		Medidas
Distritos	Regiones INEC	Muertes
-Todas las provincias	+Todas las regiones del INEC	18.183
+1 San José	-Todas las regiones del INEC	6.813
	+1 Área Metropolitana	5.327
	+2 Resto Central	938
	+3 Chorotega	
	+4 Pacífico Central	
	+5 Brunca	548
	+6 Huetar Atlántica	
	+7 Huetar Norte	
+2 Alajuela	-Todas las regiones del INEC	3.312
	+1 Área Metropolitana	
	+2 Resto Central	2.365
	+3 Chorotega	165
	+4 Pacífico Central	105
	+5 Brunca	
	+6 Huetar Atlántica	
	+7 Huetar Norte	677
+3 Cartago	-Todas las regiones del INEC	1.737
	+1 Área Metropolitana	
	+2 Resto Central	1.737
	+3 Chorotega	
	+4 Pacífico Central	
	+5 Brunca	
	+6 Huetar Atlántica	
	+7 Huetar Norte	
+4 Heredia	-Todas las regiones del INEC	1.684
	+1 Área Metropolitana	
	+2 Resto Central	1.526
	+3 Chorotega	
	+4 Pacífico Central	
	+5 Brunca	
	+6 Huetar Atlántica	57
	+7 Huetar Norte	101
+5 Guanacaste	-Todas las regiones del INEC	1.376
	+1 Área Metropolitana	
	+2 Resto Central	
	+3 Chorotega	1.376
	+4 Pacífico Central	
	+5 Brunca	
	+6 Huetar Atlántica	
	+7 Huetar Norte	
+6 Puntarenas	-Todas las regiones del INEC	1.598
	+1 Área Metropolitana	
	+2 Resto Central	
	+3 Chorotega	
	+4 Pacífico Central	857
	+5 Brunca	741
	+6 Huetar Atlántica	
	+7 Huetar Norte	
+7 Limón	-Todas las regiones del INEC	1.444
	+1 Área Metropolitana	
	+2 Resto Central	
	+3 Chorotega	
	+4 Pacífico Central	
	+5 Brunca	
	+6 Huetar Atlántica	1.444
	+7 Huetar Norte	

Slicer: [Año=2009]

Figura 39. Resultado de escenario 5 en JPivot

Resultados en GeoOLAP:

La Figura 40 evidencia que, al igual que en JPivot, la segunda dimensión espacial seleccionada se toma como subconjunto de la primera en el mapa, tabla y gráficos. Por lo tanto, si alguna de las regiones del INEC no forma parte de la provincia padre asociada, no se muestran valores para los casos, como ocurre con la región Chorotega y la provincia San José.



Provincia	Región INEC	Mea... Muer... Año 2009
1 San José	1 Área Metropolitana	5327.0
1 San José	2 Resto Central	938.0
1 San José	3 Chorotega	
1 San José	4 Pacífico Central	
1 San José	5 Brunca	548.0
1 San José	6 Huetar Atlántica	
1 San José	7 Huetar Norte	
2 Alajuela	3 Chorotega	165.0
2 Alajuela	4 Pacífico Central	105.0
2 Alajuela	5 Brunca	
2 Alajuela	6 Huetar Atlántica	
2 Alajuela	7 Huetar Norte	677.0
2 Alajuela	1 Área Metropolitana	
2 Alajuela	2 Resto Central	2365.0
3 Cartago	1 Área Metropolitana	
3 Cartago	2 Resto Central	1737.0
3 Cartago	3 Chorotega	
3 Cartago	4 Pacífico Central	
3 Cartago	5 Brunca	
3 Cartago	6 Huetar Atlántica	
3 Cartago	7 Huetar Norte	
4 Heredia	1 Área Metropolitana	
4 Heredia	2 Resto Central	1526.0
4 Heredia	3 Chorotega	
4 Heredia	4 Pacífico Central	
4 Heredia	5 Brunca	
4 Heredia	6 Huetar Atlántica	57.0
4 Heredia	7 Huetar Norte	101.0
5 Guanacaste	1 Área Metropolitana	
5 Guanacaste	2 Resto Central	
5 Guanacaste	3 Chorotega	1376.0
5 Guanacaste	4 Pacífico Central	
5 Guanacaste	5 Brunca	
5 Guanacaste	6 Huetar Atlántica	
5 Guanacaste	7 Huetar Norte	
6 Puntarenas	1 Área Metropolitana	
6 Puntarenas	2 Resto Central	
6 Puntarenas	3 Chorotega	
6 Puntarenas	4 Pacífico Central	857.0
6 Puntarenas	5 Brunca	741.0
6 Puntarenas	6 Huetar Atlántica	
6 Puntarenas	7 Huetar Norte	
7 Limón	1 Área Metropolitana	
7 Limón	2 Resto Central	
7 Limón	3 Chorotega	
7 Limón	4 Pacífico Central	
7 Limón	5 Brunca	
7 Limón	6 Huetar Atlántica	1444.0
7 Limón	7 Huetar Norte	

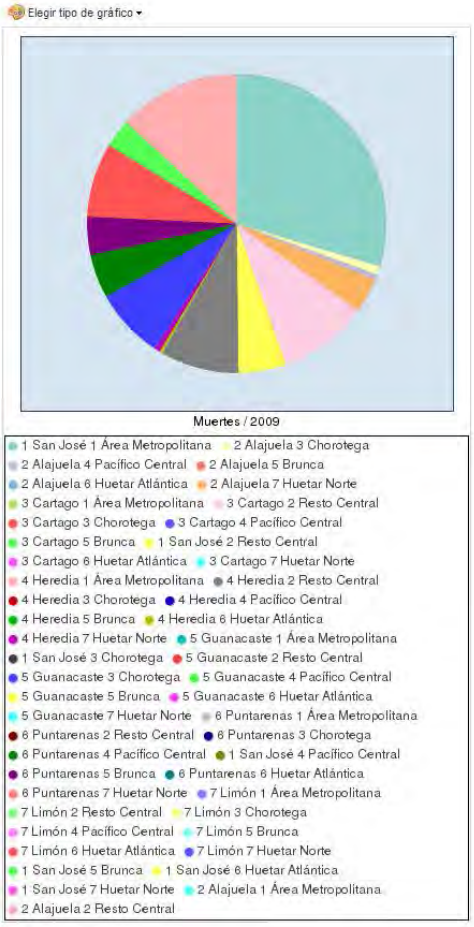


Figura 40. Resultado de escenario 5 en GeoOLAP

ESCENARIO 6 – USO EXCLUSIVO DE DIMENSIONES CONVENCIONALES

JPivot ofrece la posibilidad de crear consultas que no incluyan miembros de la dimensión espacial Distritos y sus jerarquías. En cambio, GeoOLAP, debido a su propósito inherente de análisis espacial, no permite ejecutar una consulta si no se ha seleccionado al menos un componente espacial. Por ejemplo, se puede generar una consulta para las muertes ocasionadas por cáncer de próstata durante el año 2000 para los diferentes grupos de edad, mediante los siguientes parámetros:

- Dimensiones:
    - Tiempo (2000).
    - Causa (Cáncer de Próstata).
    - Grupo especial de edad (todos los miembros).
- Medida:
    - Muertes.

Resultados en JPivot:

La Figura 41 muestra los datos resultantes luego de realizar una consulta sin incluir dimensiones espaciales.

	Medidas
	Muertes
	Tiempo
	2000
	Causa
Edad	Cáncer Próstata
-Todas las edades	295
+0-14 años	0
+15-64 años	17
+65 años y más	278
+Ignorado	

Slicer:

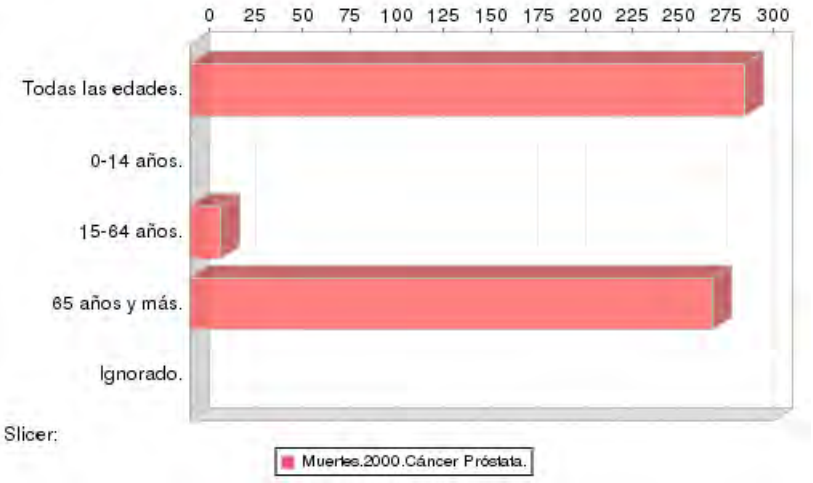


Figura 41. Resultado de escenario 6 en JPivot

ESCENARIO 7 – USO SIMULTÁNEO DE MEDIDAS

En JPivot es posible incluir varias medidas en una misma consulta. Por el contrario, GeoOLAP no tiene incorporada esta funcionalidad. Así, una consulta para obtener el número de casos y defunciones causadas por cáncer durante el año 2000 en cada una de las regiones del CCP es únicamente posible en JPivot, utilizando los siguientes parámetros:

- Dimensiones:
    - Distritos (todos los miembros del nivel Provincias).
    - Tiempo (2000).
- Medidas:
    - Casos.
    - Muertes.

Resultados en JPivot:

La Figura 42 ilustra la representación de dos medidas en una tabla y un gráfico. Esto demuestra la posibilidad de realizar consultas que incluyan más de una medida.

	Medidas	
	Muertes	Casos
	Tiempo	Tiempo
Distritos	2000	2000
-Todas las provincias	3.178	6.381
+1 San José	1.318	2.802
+2 Alajuela	549	1.141
+3 Cartago	383	656
+4 Heredia	245	564
+5 Guanacaste	214	311
+6 Puntarenas	234	535
+7 Limón	226	372

Slicer:

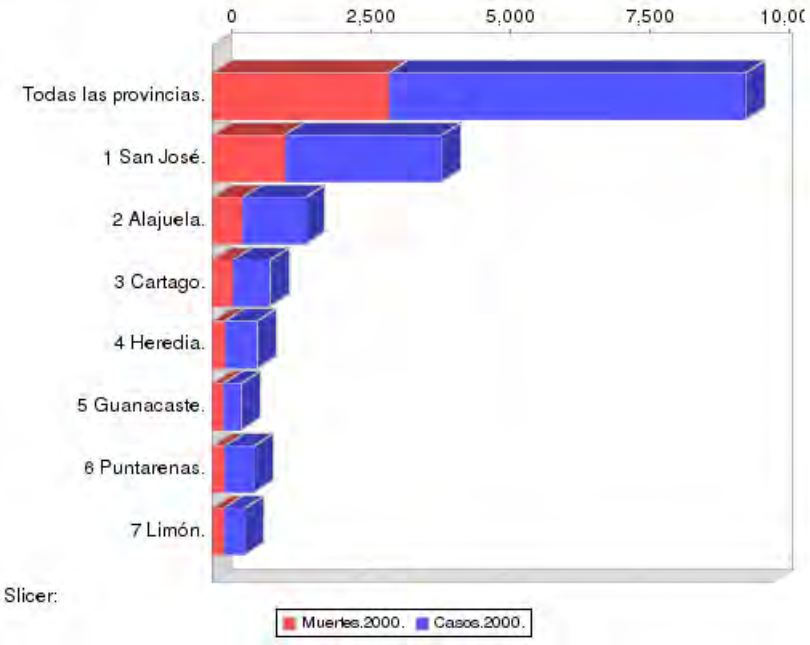


Figura 42. Resultado de escenario 7 en JPivot

2. PROBLEMAS ENCONTRADOS Y LIMITACIONES DE GEOOLAP

Inicialmente, se utilizó un servidor de prueba para evaluar los requisitos mínimos del servidor donde se instalaría GeoOLAP. Durante ese proceso, se identificó un problema de memoria: por una parte, las dimensiones de los cubos requerían cierto espacio para su carga inicial, y por otra, los miembros de las dimensiones necesitaban el suyo. Como la cantidad de memoria disponible para los miembros era menor a la necesitada, el sistema eliminaba la carga inicial de las dimensiones para crear suficiente espacio libre. Esa continua eliminación de datos en caché provocó un conflicto entre ambas cargas que afectaba el funcionamiento de la herramienta al impedir el procesamiento de consultas. No obstante, gracias a esta situación, se pudo identificar la relevancia de contar con una cantidad importante de memoria en el servidor que el CCP adquiriría exclusivamente para GeoCR, cuyas especificaciones se muestran en el Anexo F.

En relación con los problemas propios de la interfaz, los resultados que involucran muchos elementos en GeoOLAP causan que la simbología de los gráficos ocupe más espacio y, por lo tanto, complican la visualización de los gráficos. Este inconveniente se puede apreciar en la Figura 43.

Figura 43. Gráfico ocultado por la cantidad de elementos en la simbología

Además, las cantidades que aparecen como resultados en la tabla de GeoOLAP cuentan con un formato fijo: se despliegan con dos decimales. Esa representación se puede ajustar por medio del uso de un formato para la medida, que se especifica dentro de la definición de los cubos en GeoMondrian. Sin embargo, GeoOLAP no lo interpreta correctamente al mostrar las cantidades y despliega los resultados solamente con dos decimales. Esta limitación puede generar confusiones, puesto que existe el riesgo de que los usuarios no distingan si el punto en medio de los números es para destacar el millar o indicar la presencia de decimales.

Por otra parte, la combinación de dos dimensiones en GeoOLAP sólo funciona en algunos casos, mientras que la de tres dimensiones no genera resultados. Se presume que esto sucede

por el uso de ciertas geometrías que, debido a su complejidad, producen un fallo al momento de intentar hacer la intersección entre las dimensiones. En el Anexo D, se muestra el detalle de algunas dimensiones que enfrentan problemas al combinarse entre sí.

Durante esta investigación, se encontraron herramientas cliente SOLAP que cuentan con capacidades de navegación OLAP de múltiples opciones, pero que carecen de visualización en mapas; mientras tanto, otras herramientas ofrecen la opción de mostrar resultados en mapas, pero no con todas las operaciones y funcionalidades OLAP requeridas para un análisis robusto. GeoOLAP fue la opción con un mejor balance entre ambos requerimientos, ya que soporta algunas de las operaciones OLAP básicas y maneja datos espaciales; sin embargo, eso no implica que su implementación satisfaga las necesidades de los usuarios. Sus limitaciones más evidentes fueron (1) la ausencia de una opción para utilizar más de una medida por consulta y (2) la incapacidad de ejecutar consultas que no contienen dimensiones espaciales.

CAPÍTULO VIII: CONCLUSIONES Y TRABAJO FUTURO

1. CONCLUSIONES

En las últimas décadas, el uso del concepto de BI se ha extendido más allá del ámbito de los negocios, y el caso del CCP es una prueba de ello. El proyecto realizado muestra que, a través del aprovechamiento de datos existentes, es posible realizar análisis que permitan la identificación de patrones. Sistemas como GeoCR tienen potencial para ayudar en la toma de decisiones de diferentes entidades, por lo que su utilización puede beneficiar a la comunidad y al país en general.

Los almacenes de datos son elementos básicos para las soluciones BI. Ellos permiten el almacenamiento de datos históricos relacionados con los enfoques de análisis que se consideren relevantes, así como la integración de datos de distintos sistemas operacionales y fuentes externas. Para implementar un almacén, es necesario elaborar los diseños conceptual, lógico y físico. Además, se requieren procesos de ETL para aplicar las transformaciones necesarias sobre los datos y realizar su posterior carga en el almacén.

El acceso a los datos se realiza mediante un servidor SOLAP, que incluye el procesamiento de consultas y la definición de los cubos. En GeoCR, se requirió profundizar en la implementación de otras estructuras más avanzadas, como las dimensiones compartidas, medidas calculadas y los cubos virtuales. Para tal propósito, se seleccionó GeoMondrian, cuyo lenguaje de consulta es MDX. Además, la herramienta contiene una serie de funciones específicas que permiten sacar provecho de los datos espaciales.

Los resultados que genera el servidor SOLAP deben ser mostrados al usuario por un componente que contenga opciones de interfaz adaptables a datos espaciales. Para esta función, se eligió la herramienta cliente SOLAP GeoOLAP que tiene la capacidad de desplegar los resultados en tablas, gráficos y mapas de forma simultánea. Además, permite ejecutar las operaciones OLAP a través de acciones directas sobre el mapa e incluso brinda la posibilidad de combinar dimensiones espaciales.

Una vez que se tuvieron los datos integrados con las herramientas de procesamiento y visualización, se procedió a validar el adecuado funcionamiento del sistema, así como la veracidad de los datos resultantes. Para tal propósito, se crearon escenarios de análisis, los cuales muestran las operaciones disponibles en acción y determinan las posibilidades de mejora existentes para cada una de las herramientas. Asimismo, se encontraron algunos problemas relativos al *software* seleccionado, donde destacan las limitaciones de interfaz en GeoOLAP. No obstante, a pesar de que se encontraron inconvenientes durante el desarrollo del trabajo, se demostró que es posible afrontar variadas necesidades de análisis mediante de las soluciones seleccionadas.

Las herramientas SOLAP libres son escasas y demandan cambios puntuales en su contenido, de manera que se requiere un conocimiento avanzado en programación para poder hacer que funcionen. Sin embargo, el proyecto de GeoCR expone la posibilidad de brindar nuevas opciones de análisis a los usuarios no expertos, quienes, tras observar los beneficios que se les ofrecen, podrían considerar el ajuste de una solución espacial de BI *open-source* a sus

requerimientos particulares e incorporarla como un elemento importante en el proceso de toma de decisiones.

El desarrollo de este proyecto contribuyó en gran medida a la producción del artículo titulado *Spatial Data Warehouses and SOLAP Using Open-Source Tools* (incluido en el Anexo I), que fue presentado en la Conferencia Latinoamericana en Informática celebrada en octubre del 2013 en Vargas, Venezuela. Esa publicación propone el uso de un caso práctico para abordar los diferentes desafíos que pueden surgir al trabajar con herramientas SOLAP y describe posibles soluciones de software libre para implementar ese tipo de aplicaciones.

GeoCR se encuentra en línea en el sitio web del CCP, con acceso libre a sus cuatro enfoques principales: cáncer, defunciones, nacimientos y defunciones infantiles.

2. TRABAJO FUTURO

Una tarea importante del proyecto recayó en la necesidad de solucionar problemas relativos a la calidad de los datos espaciales. Con base en esa experiencia, se recomienda que las organizaciones respectivas trabajen en la adopción de estándares de calidad para sus datos. De esta forma, se evitarían inconvenientes que podrían ponerse en evidencia durante la implementación de herramientas SOLAP en general. Se presume, por ejemplo, que las inconsistencias encontradas al combinar dimensiones espaciales podrían ser resueltas mediante el refinamiento de los datos espaciales. No obstante, es importante recalcar que cualquier esfuerzo en este sentido debe estar enfocado hacia la implementación de requisitos mínimos generales para los datos y no en requerimientos particulares para un sistema específico.

En el caso particular de este proyecto, investigadores del CCP aplicaron el método de armonización de nodos para disminuir la cantidad de inconsistencias en los datos espaciales; sin embargo, al culminar el desarrollo de GeoCR, las geometrías todavía presentaban imperfecciones. Por lo tanto, se recomienda utilizar procedimientos más exhaustivos que reduzcan aún más el nivel de detalle irrelevante para el funcionamiento de la herramienta. Esto permitiría obtener geometrías más sencillas y, por consiguiente, menos susceptibles a problemas de intersección entre ellas. Otra opción consistiría en desarrollar una herramienta dedicada a realizar las verificaciones y correcciones necesarias sobre los datos espaciales en forma automática.

En relación con el procesamiento y despliegue de los datos, las mejoras pueden ser fácilmente incorporadas al sistema, ya que, al ser GeoOLAP una herramienta de *software* libre, es posible modificar su código para incluir funcionalidad de la que actualmente carece. Por ejemplo, sería de utilidad agregar la totalidad de las características de JPivot en GeoOLAP; las posibilidades de análisis incrementarían con la incorporación de la operación *pivot*, la selección de varias medidas en una misma consulta y la opción de usar la tabla para hacer *drill-down* en varios miembros de dimensiones. Asimismo, sería útil añadir operaciones inherentemente espaciales dentro de la herramienta SOLAP cliente (por ejemplo, el cálculo del área), las cuales permitirían el procesamiento de medidas como "casos por kilómetro cuadrado". La ausencia de estas características en GeoOLAP está determinada por las limitaciones de su interfaz; por lo tanto, habría que realizar un trabajo de depuración en la misma, donde se implementen nuevos métodos para proyectar lo requerido en el mapa.

Por otra parte, GeoOLAP maneja un espacio de tamaño fijo para desplegar los gráficos junto a su simbología, lo cual causa de que se muestre sólo la simbología cuando existen demasiados elementos en ella. Esa situación —mencionada en la sección de problemas encontrados— deja entrever la necesidad de utilizar parámetros en todos los espacios de despliegue (mapa, tabla y gráficos), de manera que el sistema sea flexible y se adapte a la cantidad de elementos que debe mostrar. De lo contrario, existe el peligro de que el usuario se encuentre con elementos que no le permitan ver la representación de los datos. Ese mismo problema ocurre cuando se descarga un archivo PDF, por lo que también se podría trabajar para mejorar su formato.

La visualización constituye un aspecto de gran importancia en el sistema, pero al mismo tiempo es de los que más deficiencias presentan. Algunas de las limitaciones de GeoOLAP fueron corregidas durante el curso del proyecto; sin embargo, todavía existe un espacio de mejora con respecto a la interfaz, el cual sería importante explorar en trabajos futuros.

Aunque en el presente trabajo se mencionan algunos beneficios derivados de la utilización de GeoCR y sus componentes, es importante conocer la opinión de los usuarios sobre la herramienta. Por consiguiente, es necesario investigar la percepción que tienen los usuarios del sistema desarrollado en comparación con otras herramientas de consulta que ofrece el CCP.

REFERENCIAS

[ALB99] Albrecht, J. y otros. "Management of multidimensional aggregates for efficient online analytical processing". *IDEAS '99. International Symposium Proceedings Database Engineering and Applications.* Montral, Canada: IEEE, 1999. 156-164.

[ASA12] *Asamblea Legislativa de la República de Costa Rica.* http://www.asamblea.go.cr/Centro_de_Informacion/Consultas_SIL/Pginas/Detalle%20Proyectos%20de%20Ley.aspx?Numero_Proyecto=16912 (último acceso: 9 de Junio de 2012).

[BED01] Bédard, Yvan, Tim Merrett, y Han Jiawei. "Fundamentals of spatial data warehousing for geographic knowledge discovery". En *Geographic Data Mining and Knowledge Discovery*, de Harvey Milller y Han Jiawei, 53-73. London: Chapman & Hall, 2001.

[BED09] Bédard, Yvan, Eveline Bernier, Suzie Larrivée, Martin Nadeau, Marie-Josée Proulx, y Sonia Rivest. *Spatial OLAP.* November de 2009. http://www.spatialbi.com (último acceso: 20 de August de 2013).

[BED98] Bedell, J. A. "Outstanding challenges in OLAP". *14th International Conference on Data Engineering.* Orlando, FL, US: IEEE, 1998. 178-179.

[BIM07] Bimonte, Sandro, Anne Tchounikine, y Maryvonne Miquel. “Spatial OLAP: Open Issues and a Web Based Prototype”. *10th AGILE International Conference on Geographic Information Science.* Aalborg, Denmark, 2007.

[CEN12] *Centro Centroamericano de Población.* http://ccp.ucr.ac.cr/censos (último acceso: 28 de Agosto de 2012).

[CEN13] *Centro Centroamericano de Población - InfoCensos.* http://infocensos.ccpucr.ucr.ac.cr/Mapas.43.0.html (último acceso: 14 de Octubre de 2013).

[CHE08] Chen, Rongguo, y Jion Xie. “Open Source Databases and Their Spatial Extensions”. En *Open Source Approaches in Spatial Data Handling*, 105-130. Springer, 2008.

[DYR01] Dyreson, Curtis E., Christian S. Jensen, y Torben Bach Pedersen. “A Foundation for Capturing and Querying Complex Multidimensional Data”. *Information Systems* 26, nº 5 (2001): 383-423.

[ENV13] Environmental Systems Research Institute. “ESRI Shapefile Technical Description - An ESRI White Paper”. Julio de 1998. http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf (último acceso: 26 de Marzo de 2013).

[GIA08] Giacometti, Arnaud, Patrick Marcel, y Elsa Negre. “A framework for recommending OLAP queries”. *DOLAP '08 Proceedings of the ACM 11th international workshop on Data warehousing and OLAP.* Napa Valley, CA, US: ACM, 2008. 73-80.

[GIA09] Giacometti, Arnaud, Patrick Marcel, Elsa Negre, y Arnaud Soulet. “Query recommendations for OLAP discovery driven analysis”. *DOLAP '09 Proceedings of the ACM twelfth international workshop on Data warehousing and OLAP.* New York, NY, USA: ACM, 2009. 81-88.

[GUI12] Guide, Oracle9i Data Warehousing. *Schema Modeling Techniques.* http://docs.oracle.com/cd/B10501_01/server.920/a96520/schemas.htm (último acceso: 13 de 06 de 2012).

[HOS08] Hose, Katja, Daniel Klan, Matthias Marx, y Kai-Uwe Sattler. “When is it time to rethink the aggregate configuration of your OLAP server?» *Proceedings of the VLDB Endowment*, 2008: 1492-1495.

[HOS09] Hose, Katja, Daniel Klan, y Kai-Uwe Sattler. “Online Tuning of Aggregation Tables for OLAP”. *ICDE '09. IEEE 25th International Conference on Data Engineering.* Shanghai, China: IEEE, 2009. 1679-1686.

[HSI11] Hsiao, Tim, Wo-Shun Luk, y Stephen Petchulat. “Data visualization on web-based OLAP”. *DOLAP '11 Proceedings of the ACM 14th international workshop on Data Warehousing and OLAP.* Glasgow, Scotland, UK: ACM, 2011. 75-82.

[HUY01] Huyn, Nam. “Scientific OLAP for the Biotech Domain”. *VLDB '01 Proceedings of the 27th International Conference on Very Large Data Bases.* San Francisco, CA, US: Morgan Kaufmann Publishers Inc, 2001. 645-648.

[INM96] Inmon, W. H. “The data warehouse and data mining”. *Communications of the ACM* 39, nº 11 (1996): 49-50.

[INM05] Inmon, W. H. *Building the data warehouse.* 3rd. New York: Wiley Computer Publishing, 2005.

[INS13] Institute for Statistics and Mathematics. *The R Project for Statistical Computing.* s.f. http://www.r-project.org/ (último acceso: 2 de Noviembre de 2013).

[JEN01] Jensen, Mikael R., Thomas H. Moller, y Torben Bach Pedersen. “Specifying OLAP Cubes on XML Data”. *Proceedings Thirteenth International Conference on Scientific and Statistical Database Management.* Fairfax, VA , US: Kluwer Academic Publishers Hingham, 2001. 101-112.

[JUK06] Jukic, Nenad. “Modeling strategies and alternatives for data warehousing projects”. *Communications of the ACM* 49, nº 4 (2006): 83-88.

[KIM11] Kimball, Ralph, y Margy Ross. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling.* 2. New York: John Wiley & Sons, 2011.

[KOP09] Kopáčková, Hana, y Markéta Škrobáčková. “Decision Support Systems or Business Intelligence: What can help in decision making?» *Digitální knihovna Univerzity Pardubice.* 03 de September de 2009. http://dspace.upce.cz/bitstream/10195/32436/1/CL585.pdf (último acceso: 20 de April de 2012).

[LAW05] Lawrence, Ramon, y Anton Kruger. “An Architecture for Real-Time Warehousing of Scientific Data”. *International Conference on Scientific Computing.* Las Vegas, Nevada, US, 2005. 151-156.

[LEA12] Lean, David. *SQL 2008 Spatial Samples, Part 2 of 9 - Background on Spatial Types & Well Known Text (WKT).* http://blogs.msdn.com/b/davidlean/archive/2008/11/01/sql-2008-spatial-samples-part-2-of-n-background-on-spatial-types-well-known-text-wkt.aspx (último acceso: 14 de 06 de 2012).

[LIU02] Liu, Wen-Yuan, y Shu-Fen Fang. “OLAP realization technology research based on MDX”. *2002 International Conference on Machine Learning and Cybernetics.* Beijing, China: IEEE, 2002. 2205-2209.

[MAL08] Malinowski, Elzbieta, y Esteban Zimányi. *Advanced data warehouse design: from conventional to spatial and temporal applications.* Nueva York: Springer, 2008.

[MOR07] Morera, Melvin, y Amada Aparicio. “Diferencias territoriales en el funcionamiento de las áreas de salud: variabilidad geográfica de las hospitalizaciones evitables y factores asociados”. 2007. http://www.estadonacion.or.cr/images/stories/informes/013/docs/Equidad/Aparicio-Morera-2007.pdf (último acceso: 05 de Enero de 2013).

[MSD03] MSDN Library, Microsoft. *Consulta de MDX básica (MDX).* Microsoft MSDN Library. http://msdn.microsoft.com/es-es/library/ms144785.aspx (último acceso: 26 de Marzo de 2013).

[MSD07] MSDN Library, Microsoft. *Virtual Cubes.* http://msdn.microsoft.com/en-us/library/aa216377%28v=sql.80%29.aspx (último acceso: 01 de Julio de 2013).

[NAT08] National Geospatial-Intelligence Agency. *Earth Gravitational Model EGM2008.* s.f. http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm2008/index.html (último acceso: 22 de Octubre de 2013).

[NIE01] Niemi, Tapio, Jyrki Nummenmaa, y Peter Thanisch. “Constructing OLAP cubes based on queries”. *DOLAP '01: Proceedings of the 4th ACM international workshop on Data warehousing and OLAP.* Atlanta, GA, US: ACM, 2001. 22-27.

[NIE02] Niemi, Tapio, Marko Niinimäki, Jyrki Nummenmaa, y Peter Thanisch. “Constructing an OLAP cube from distributed XML data”. *DOLAP '02: Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP.* McLean, VA, US: ACM, 2002. 22-27.

[PAR08] Pardillo, Jesús, Jose-Norberto Mazón, y Juan Trujillo. “Bridging the semantic gap in OLAP models: platform-independent queries”. *DOLAP '08 Proceedings of the ACM 11th international workshop on Data warehousing and OLAP.* Napa Valley, CA, US: ACM, 2008. 89-96.

[PED02] Pedersen, Dennis, Torben Bach Pedersen, y Karsten Riis. “Query optimization for OLAP-XML federations”. *DOLAP '02: Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP.* McLean, VA, US: ACM, 2002. 57-64.

[PED00] Pedersen, Torben Bach, y Christian S. Jensen. “Advanced Implementation Techniques for Scientific Data Warehouses”. *Proceedings of the Workshop of Management and Integration of Biochemical Data.* Villa Bosch, Heidelberg, Germany: IEEE, 2000. 1-9.

[POS03] PostGIS. *About PostGIS.* http://postgis.net/ (último acceso: 2013 de Marzo de 2013).

[POS04] —. *PostGIS 2.0 Manual.* http://postgis.net/docs/manual-2.0/index.html (último acceso: 11 de Abril de 2013).

[PRE14] Presidencia de la República de Costa Rica. «Reglamento de Especificaciones para la Delimitación de la zona pública de la Zona.» Sistema Nacional de Áreas de Conservación. s.f. http://www.sinac.go.cr/normativa/Decretos/Reglamento%20de%20Especificaciones%20para%20la%20Delimitaci%C3%B3n%20de%20la%20zona%20p%C3%BAb%20de%20la%20Zona%20Mar%C3%ADtimo%20Terrestre%20N%C2%BA%2036642.pdf (último acceso: 06 de Febrero de 2014).

[RIV01] Rivest, S., Y. Bédard, y P. Marchand. "Towards better support for spatial decision-making: defining the characteristics of Spatial On-Line Analytical Processing (SOLAP)". *Geomatica, the Journal of the Canadian Institute of Geomatics* 55, nº 4 (2001): 539–555.

[RIV03] Rivest, S., Y. Bédard, M.J. Proulx, y M. Nadeau. "SOLAP: a new type of user interface to support spatio-temporal multidimensional data exploration and analysis". *Proceedings of the ISPRS Joint Workshop on Spatial, Temporal and Multi-Dimensional Data Modelling and Analysis.* Quebec, Canada, 2003. 45-56.

[RIV05] Rivest, Sonia, Yvan Bédard, Marie-Josée Proulx, Martin Nadeau, Frederic Hubert, y Julien Pastor. “SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data”. *ISPRS Journal of International Society for Photogrammetry and Remote Sensing* 1, nº 60 (2005): 17-33.

[ROS02] Rosero, Luis. “Estimaciones y proyecciones de población por distrito y otras áreas geográficas. Costa Rica 1970-2015”. Centro Centroamericano de Población, Universidad de Costa Rica, Instituto Nacional de Estadística y Censos. s.f. http://ccp.ucr.ac.cr/bvp/pdf/proye/distrital.pdf (último acceso: 05 de Enero de 2013).

[SAL08] Salam, Ismail, Mohammed El Mohajir, Abdeslam Taleb, y Badreddine El Mohajir. “Development of SOLAP patrimony management application system: Fez medina as a case study”. *International Journal of Computer Science and Applications*, 2008: 57-66.

[SCO05] Scotch, Matthew, y Parmanto, Bambang. “SOVAT: Spatial OLAP Visualization and Analysis Tool”. *HICSS '05 Proceedings of the 38th Annual Hawaii International Conference on System Sciences.* Big Island, 2005. 142b.

[SOU04] Sourceforge. *JPivot.* sourceforge. http://jpivot.sourceforge.net/ (último acceso: 30 de Abril de 2013).

[SPA13] *SpagoBI - BI components.* http://www.spagoworld.org/xwiki/bin/view/SpagoBI/BIComponents (último acceso: 20 de August de 2013).

[SPA22] Spatialytics.org. *GeoMondrian - Spatialytics.org SOLAP Server.* s.f. http://www.spatialytics.org/projects/GeoMondrian/ (último acceso: 22 de Setiembre de 2013).

[TOL99] Tolkin, Steven. "Aggregation everywhere: data reduction and transformation in the Phoenix data warehouse". *DOLAP '99 Proceedings of the 2nd ACM international workshop on Data warehousing and OLAP.* Kansas City, MO, US: ACM, 1999. 79-86.

[TSO03] Tsois, Aris, y Timos Sellis. "The Generalized Pre-Grouping Transformation: Aggregate-Query Optimization in the Presence of Dependencies". *VLDB '03 Proceedings of the 29th international conference on Very large data bases.* Berlin, Germany: ACM, 2003. 644-655.

[ZHE10] Zhenyuan, Wu, y Hu Haiyan. "OLAP Technology and Its Business Application". *2010 Second WRI Global Congress on Intelligent Systems (GCIS).* Wuhan, China: IEEE, 2010. 92-95.

ANEXOS

ANEXO A – CREACIÓN DEL ALMACÉN DE DATOS

Antes de crear el almacén de datos, se instaló PostgreSQL y su extensión espacial PostGIS. En el caso de GeoCR, la instalación se realizó en Ubuntu siguiendo estos pasos:

1. Actualización del repositorio apt-get:

```
apt-get update
```

2. Descarga de PostgreSQL:

```
sudo apt-get install postgresql postgresql-contrib
```

3. Instalación de PostGIS:

```
sudo apt-get install python-software-properties
sudo apt-add-repository ppa:sharpie/for-science
sudo apt-add-repository ppa:sharpie/postgis-stable
sudo apt-add-repository ppa:ubuntugis/ubuntugis-unstable
sudo apt-get update
sudo apt-get install postgis
```

Una vez completada la instalación, se creó la base de datos y se le agregó la extensión espacial (PostGIS):

1. Creación de la base de datos:

```
createdb almacen_datos_cc
```

2. Habilitación del lenguaje PL/pgSQL en la nueva base de datos (es necesario porque muchas de las funciones de PostGIS están escritas en ese lenguaje):

```
createlang plpgsql almacen_datos_ccp
```

3. Carga de los objetos y definiciones propias de PostGIS en la base de datos:

```
psql -d almacen_datos_ccp -f
/usr/share/postgresql/9.1/contrib/postgis-2.0/postgis.sql
```

4. Carga de los sistemas de coordenadas a la base de datos:

```
psql -d almacen_datos_ccp -f
/usr/share/postgresql/9.1/contrib/postgis-
2.0/spatial_ref_sys.sql
```

Con la base de datos creada, se procedió a crear las tablas mediante las siguientes sentencias SQL:

```
CREATE TABLE edades_especial (
    id integer NOT NULL,
    nombre_edad character varying(50),
    codigo_edad integer,
    codigo_grupo_especial integer,
    codigo_grupo_grande integer,
    nombre_grupo_especial character varying(50),
    nombre_grupo_grande character varying(50)
);


CREATE TABLE categorias_especial (
    id integer NOT NULL,
    codigo integer,
    nombre character varying(110),
    codigo_categoria integer,
    nombre_categoria character varying(50)
);


CREATE TABLE sexo (
    codigo smallint PRIMARY KEY,
    nombre character varying(50),
);


CREATE TABLE geografia_especial2 (
    id integer,
    codigo_distrito integer,
    nombre_distrito character varying(50),
    codigo_canton integer,
```

```
    nombre_canton character varying(50),
    codigo_provincia integer,
    nombre_provincia character varying(50),
    codigo_inec integer,
    nombre_inec character varying(50),
    codigo_ccp integer,
    nombre_ccp character varying(50),
    codigo_sub_inec integer,
    nombre_sub_inec character varying(50),
    codigo_urbano integer,
    nombre_urbano character varying(50),
    codigo_area_ccss integer,
    nombre_area_ccss character varying(50),
    codigo_region_ccss integer,
    nombre_region_ccss character varying(50),
    codigo_gam integer,
    nombre_gam character varying(50),
    geometria_distrito geometry,
    geometria_canton geometry,
    geometria_provincia geometry,
    geometria_area_ccss geometry,
    geometria_region_ccss geometry,
    geometria_inec geometry,
    geometria_sub_inec geometry,
    geometria_urbano geometry,
    geometria_gam geometry,
    geometria_pais geometry,
    nombre_pais character varying(50),
    geometria_ccp geometry
);


CREATE TABLE poblacion (
    anno integer,
    distrito integer,
    sexo integer,
    edad smallint,
    poblacion integer
);


CREATE TABLE cancer_no_desconocidos (
    id integer NOT NULL,
    distrito integer references geografia_especial2(codigo_distrito),
    anno integer,
    sexo smallint references sexo(codigo),
    localiz integer references categoria_especial(id),
    muertes integer,
    edad integer references edades_especial(codigo_edad),
    poblacion integer,
    casos integer
);
```

```
CREATE TABLE defunciones_no_desconocidos (
    id integer NOT NULL,
    distrito integer references geografia_especial2(codigo_distrito),
    anno integer,
    sexo smallint references sexo(codigo),
    causa integer references categoria_especial(id),
    muertes integer,
    edad integer references edades_especial2(codigo_edad),
    poblacion integer
);


CREATE TABLE defunciones_infantiles_r (
    id integer NOT NULL,
    anno integer,
    sexo smallint references sexo(codigo),
    muertes integer,
    distrito integer references geografia_especial2(codigo_distrito),
    edad integer references horas(codigo),
    causa integer references categoria_especial(id)
);


CREATE TABLE nacimientos (
    id integer NOT NULL,
    anno integer,
    sexo smallint references sexo(codigo),
    nacimientos integer,
    distrito integer references geografia_especial2(codigo_distrito),
);
```

ANEXO B – FUNCIONES ESPACIALES DE GEOMONDRIAN

Tabla B.1. Funciones espaciales de GeoMDX[12]

Función y parámetros	Descripción de resultado obtenido
ST_Area (Geometry)	El área de una geometría.
ST_Buffer (Geometry, Number)	El buffer (con la distancia especificada) alrededor de una geometría.
ST_Centroid (Geometry)	El centroide de una geometría.
ST_Contains (Geometry, Geometry)	Verdadero si la primera geometría contiene a la segunda en su totalidad.
ST_ConvexHull (Geometry)	La envolvente convexa de una geometría.
ST_Crosses (Geometry, Geometry)	Verdadero si la primera geometría cruza a la segunda.
ST_Difference (Geometry, Geometry)	Una geometría que representa la parte de la primera geometría que no es parte de la segunda.
ST_Disjoint (Geometry, Geometry)	Verdadero si la primera geometría es disjunta de la segunda (no la interseca).
ST_Distance (Geometry, Geometry)	La distancia entre las dos geometrías.
ST_Envelope (Geometry)	El rectángulo delimitador mínimo (MBR) de una

[12] Trac Repository Browser, "GeoMondrian, Spatialytics.org SOLAP Server". [En línea]. Disponible: http://trac.spatialytics.com/GeoMondrian/browser/tags/1.0/src/main/mondrian/udf/geo [Último acceso: 27 de junio, 2013]

	geometría.
ST_Equals (Geometry, Geometry)	Verdadero si la primera geometría es igual a la segunda.
ST_GeomFromText (String)	Una geometría creada a partir de una hilera en formato WKT.
ST_Intersection (Geometry, Geometry)	Una geometría que es la parte común de las dos geometrías.
ST_Intersects (Geometry, Geometry)	Verdadero si la primera geometría interseca a la segunda.
ST_Length (Geometry)	La longitud de una geometría.
ST_Overlaps (Geometry, Geometry)	Verdadero si la primera geometría se traslapa con la segunda.
ST_Relate (Geometry, Geometry, String)	El resultado de la relación espacial especificada por la matriz del modelo dimensional extendido de 9 intersecciones (DE-9IM), aplicada a las dos geometrías.
ST_SymDifference (Geometry, Geometry)	Una geometría que es la diferencia simétrica de dos geometrías (por ejemplo, las partes que no se intersecan).
ST_Touches (Geometry, Geometry)	Verdadero si la primera geometría toca la segunda.
ST_Transform (Geometry, Number,	Una geometría reproyectada de un SRID fuente a

Number)	un SRID destino.
ST_Union (Geometry, Geometry)	Una geometría que es el conjunto de puntos de la unión de las geometrías.
ST_Within (Geometry, Geometry)	Verdadero si la primera geometría está contenida por completo en la segunda geometría.

ANEXO C – CAMBIOS EN LA INTERFAZ DE GEOOLAP

Se realizaron diversas modificaciones en la interfaz de GeoOLAP. En algunas —mostradas en la Tabla C.1— fue suficiente editar el archivo que contenía el código JavaScript para controlar la interfaz, pero otras —presentadas en la Tabla C.2— requirieron alterar los archivos Java de la herramienta en sí. En el caso de estos últimos, se debió compilar la herramienta completa para ver el resultado de cada modificación.

Tabla C.1. Cambios realizados en archivo app-all.js

Cambio propuesto	Modificación en el código
Agrandar el *combo box* en donde se escoge la medida.	Se agregó la instrucción: `tpl:'<tpl for="."> <div xt:qtip="{MEASURE_NAME}" class="x-combo-list-item"> {MEASURE_NAME}</div></tpl>'`
Agregar *tooltip* al *combo box* sobre el mapa, para seleccionar indicador en el estilo del mapa.	Se agregó la instrucción: `tpl:'<tpl for="."><div ext:qtip="{name}" class="x-combo-list-item">{name}</div></tpl>'`
Ocultar la opción para elegir entre valores absolutos o relativos.	Se eliminó la instrucción: `H.show();`

Agregar una barra de desplazamiento al escoger los niveles de las dimensiones.	Se cambió la instrucción:
	`this.menu{}`
	Por:
	`this.menu={height:200,width:200,autoScroll:true};`

Tabla C.2. Cambios realizados en el código fuente de GeoOLAP

Cambio propuesto	Modificación en el código

Seleccionar la codificación adecuada para el despliegue de caracteres especiales del idioma español.	En el archivo GetLegend.java, se modificó la codificación de las etiquetas de la simbología reemplazando la instrucción:
	`labels = URLEncoder.encode(labels, "UTF-8");`
	Por:
	`labels = URLEncoder.encode(labels, "ISO-8859-1");`

Traducir todas las hileras al idioma español.	En los casos de tildes, se utilizó Unicode. Por ejemplo, 'í' se especificó como '\u00ed'.
Agrandar la imagen que contiene la simbología de las coropletas.	En el archivo GetLegend.java, específicamente en la función `prepareLegendSize`, se cambió el tamaño inicial de la imagen —la variable llamada `height`— de 1 a 35.

Reparar el despliegue de símbolos proporcionales sobre el mapa.	En el archivo GetLegend.java, se modificó el número de puerto —de 8080 a 80— en los enlaces que crean los símbolos. Por ejemplo, la instrucción:
	`URL url = new URL("http://localhost:8080/webbi/getoverlayicon?type=" + type +"&width=" + size + "&height=" + size + "&data=" + data + "&legend=false");`
	Pasó a ser:
	`URL url = new URL("http://localhost:80/webbi/getoverlayicon?type=" + type + "&width=" + size + "&height=" + size + "&data=" + data + "&legend=false");`

ANEXO D – COMBINACIÓN DE DIMENSIONES ESPACIALES

La Tabla D.1 muestra la lista de algunas dimensiones espaciales que causaron problemas al combinarlas para formar una consulta en GeoOLAP.

Tabla D.1. Combinación de dimensiones espaciales

Dimensión 1	**Dimensión 2**	**Miembros problemáticos**
Todas las provincias	Región CCP	1 Metro San José
		3 Rural Valle Central
		5 Rural Bajura
	Área de Salud	609 El Guarco (2 Central Sur)
		613 Los Santos (2 Central Sur)
		302 Bagaces (3 Chorotega)
		Todos los de 4 Pacífico Central
		505 Osa (5 Brunca)
	Subregión INEC	4 Cartago
		12 Los Santos
		19 Buenos Aires
Todas Regiones de Salud	Cantón	605 Osa (6 Puntarenas)
	Subregión INEC	603 Buenos Aires (6 Puntarenas)

Todas Regiones CCP	Provincia	1 San José
		3 Cartago
		4 Heredia
	Cantón	111 Vásquez d Corona (1 San José)
		114 Moravia (1 San José)
		117 Dota (1 San José)
		308 El Guarco (3 Cartago)
		401 Heredia (4 Heredia)
		605 Osa (6 Puntarenas)
	Área de Salud	107 Coronado (1 Central Norte)
		113 Moravia (1 Central Norte)
		120 Sta Bárbara-Varablanca (1 Central Norte)
		609 El Guarco (2 Central Sur)
		613 Los Santos (2 Central Sur)
		302 Bagaces (3 Chorotega)
		505 Osa (5 Brunca)
	GAM	Todos
	Zona	2 Periferia Urbana
		3 Rural concentrado

Todas Regiones INEC	Cantón	605 Osa (6 Puntarenas)
	Área de Salud	302 Bagaces (3 Chorotega)
		505 Osa (5 Brunca)
	Subregión INEC	603 Buenos Aires (6 Puntarenas)

ANEXO E – CÓDIGOS PARA MAPEO DE TIPOS DE CÁNCER Y CAUSAS DE MUERTE

Se requirió modificar los códigos de tipos de cáncer y causas de defunciones, de modo que pudieran ser manejados por las herramientas seleccionadas. La Tabla E.1 muestra los códigos originales y los códigos finales que se insertaron en el almacén para las causas de defunciones, mientras que la Tabla E.2 contiene la recodificación de las causas de muerte, aplicada para convertirlas en causas de muerte infantil.

Tabla E.1. Mapeo de códigos de tipo de cáncer

Tipo de cáncer	Código original CIE-10	Código original CIE-9	Código final
Tumor maligno del labio, de la cavidad bucal y de la faringe	C00-C14	140-149	9
Tumor maligno del esófago	C15	150	10
Tumor maligno del estómago	C16	151	11
Tumor maligno del colon	C18	153	12
Tumor maligno del recto, de la porción rectosigmoide y del ano	C19-C21	154	13
Tumor maligno del hígado y vías biliares intrahepáticas	C22	155	14
Tumor maligno del páncreas	C25	157	15
Otros tumores malignos digestivos	Resto C15-C26,	Resto 150-159	16

	C45.1, C48		
Tumor maligno de la laringe	C32	161	17
Tumor maligno de la tráquea, de los bronquios y del pulmón	C33, C34	162	18
Otros tumores malignos respiratorios e intratorácicos	Resto C30-C39, C45.0.2	Resto 160-165	19
Tumores malignos del hueso y de los cartílagos articulares	C40, C41	170	20
Melanoma maligno de la piel	C43	172	21
Otros tumores malignos de la piel y de los tejidos blandos	C44-C47, C49 (excepto C45.0.1.2)	171, 173	22
Tumor maligno de la mama	C50	174,175	23
Tumor maligno del cuello del útero	C53	180	24
Tumor maligno de otras partes del útero	C54, C55	179,182	25
Tumor maligno del ovario	C56	183	26
Tumores malignos de otros órganos genitales femeninos	Resto C51-C58	Resto 179-184	27
Tumor maligno de la próstata	C61	185	28
Tumores malignos de otros órganos genitales masculinos	Resto C60-C63	186,187	29

Tumor maligno del riñón, excepto pelvis renal	C64	189	30
Tumor maligno de la vejiga	C67	188	31
Otros tumores malignos de las vías urinarias	Resto C64-C68	Resto 188-189	32
Tumor maligno del encéfalo	C71	191	33
Otros tumores malignos neurológicos y endocrinos	Resto C69-C75	Resto 190-194	34
Tumor maligno de sitios mal definidos, secundarios y de sitios no especificados	C76-C80, C97	195-199	35
Tumores malignos del tejido linfático, de los órganos hematopoyéticos y de tejidos afines	C81-C90, C96	200-203, 273.3	36
Leucemia	C91-C95	204-208	37
Tumores in situ	D00-D09	230-234	38
Tumores benignos	D10-D36	210-229	39
Síndrome mielodisplásico	D46	289.8	40
Otros tumores de comportamiento incierto o desconocido	D37-D45, D47, D48	235-239, 273.1	41

Tabla E.2. Mapeo de códigos de causa de muerte

Código de causa de muerte	Etiqueta de causa de muerte	Etiqueta de causa de muerte infantil	Código de causa de muerte infantil
1	Diarrea	Diarrea	1
5	Malnutric	Malnutric	3
7	Perinatal	Perinatal	4
8	Congénita	Congénita	5
15	Resp Crónica	Resp Crónica	6
16	Cardio Vascular	Cardio Vascular	7
2	TB Resp	Resto Infecc	2
3	IRA		
4	Resto Infecc		
19	Acc Transp	Otros accidentes	8
20	Otros accidentes		
6	Materna	Residual	9
9	Cáncer de Estómago		
10	Cáncer Respiratorio		
11	Cáncer Útero		
12	Cáncer Mama		
13	Cáncer Próstata		
14	Otros cáncer		
17	Diabetes		
18	Alcohol-Cirrosis		

21	Suicidios		
22	Homicidio		
23	HIV-SIDA		
24	Residual		

ANEXO F – ESPECIFICACIONES DEL SERVIDOR ADQUIRIDO PARA LA INSTALACIÓN DE GEOCR

Tabla F.1. Especificaciones del servidor

Característica	**Especificación**
Memoria	21.39 GB
Espacio de intercambio	11.99 GB
CPU	Intel(R) Xeon(R) CPU E5645 @ 2.40 GHz 5 núcleos
Sistema operativo	Ubuntu 12.04 (64-bit)
Disco duro	200GB

ANEXO G – PASOS PARA LA INSTALACIÓN DE GEOOLAP[13]

GeoOLAP es una aplicación desarrollada en lenguaje Java, por lo que se debió instalar Java 1.7.0_05-b05 para trabajar con ella. Además, se requirió la instalación previa de un servidor HTTP que fuera capaz de ejecutar su código. Para ello, se usó la versión 7.0.29 de Apache Tomcat.

Posteriormente, se movió el archivo webbi.war de GeoOLAP al directorio de aplicaciones web de Tomcat (webapps). Ese fichero se descomprime automáticamente y genera una carpeta con lo necesario para ejecutar la herramienta. Sin embargo, en ese punto aún hace falta ajustar ciertos parámetros de manera que se puedan adaptar al caso específico.

Luego de instalar las herramientas relacionadas con el almacén de datos e insertar los datos en el mismo (como se indica en el Anexo A), se establece su vínculo con la herramienta a través del archivo ws-servlet.properties (/webapps/webbi/WEB-INF/). En él, se indican, entre otras cosas, (1) el nombre del esquema XML que contiene la definición de los cubos y (2) el nombre, usuario y contraseña del almacén de datos.

Finalmente, se agrega el esquema XML en la ruta correspondiente dentro de Tomcat (/webapps/webbi/WEB-INF/classes/) y se reinicia el servidor HTTP. Por defecto, la herramienta es accesible desde un navegador mediante el puerto 8080.

[13] Elaborados a partir de información encontrada en los repositorios de la herramienta GeoOLAP, disponibles en https://github.com/pmauduit/GeoBI y https://github.com/camptocamp/GeoBI/wiki/Install.

ANEXO H – ESQUEMA DE GEOMONDRIAN

Código G.1. Esquema completo de GeoMondrian

```
1  <Schema name="RNTwebbi">
2     <Dimension name="Sexo">
3           <Hierarchy hasAll="true" allMemberName="Ambos sexos"
           primaryKey="codigo">
4                 <Table name="sexo"/>
5                 <Level name="Sexo" column="codigo" nameColumn="nombre"
                  uniqueMembers="false"/>
6           </Hierarchy>
7     </Dimension>
8     <Dimension name="Edad">
9           <Hierarchy hasAll="true" allMemberName="Todas las edades"
           primaryKey="codigo_edad">
10                <Table name="edades_especial"/>
11                <Level name="Grupo grande"
                  column="codigo_grupo_grande"
                  nameColumn="nombre_grupo_grande"
                  uniqueMembers="false"/>
12                <Level name="Grupo especial"
```

```
                column="codigo_grupo_especial"
                nameColumn="nombre_grupo_especial"
                uniqueMembers="false"/>
13              <Level name="Edad quinquenal" column="codigo_edad"
                nameColumn="nombre_edad" uniqueMembers="false"/>
14          </Hierarchy>
15      </Dimension>
16      <Dimension name="Tipo">
17          <Hierarchy hasAll="true" allMemberName="Todos los tipos"
            primaryKey="codigo">
18              <View alias="cat_cancer">
19                  <SQL dialect="generic"><![CDATA[select * from
                    categorias_especial where codigo_categoria =
                    1]]></SQL>
20              </View>
21              <Level name="Tipo" column="codigo" nameColumn="nombre"
                uniqueMembers="false"/>
22          </Hierarchy>
23      </Dimension>
24      <Dimension name="Causa">
25          <Hierarchy hasAll="true" allMemberName="Todas las causas"
```

```
        primaryKey="id">
            <View alias="cat_defunciones">
26
                <SQL dialect="generic"> <![CDATA[select * from
                categorias_especial where codigo_categoria = 2]]>
                </SQL>
27
            </View>
28
            <Level name="Causa" column="id" nameColumn="nombre"
            uniqueMembers="false"/>
29
        </Hierarchy>
30
    </Dimension>
31
    <Dimension name="Causa_infantil">
32
        <Hierarchy hasAll="true" allMemberName="Todas las causas"
        primaryKey="id">
33
            <View alias="cat_defunciones">
34
                <SQL dialect="generic"> <![CDATA[select * from
                categorias_especial where codigo_categoria = 3]]>
                </SQL>
35
            </View>
36
            <Level name="Causa" column="id" nameColumn="nombre"
            uniqueMembers="false"/>
37
```

```
38          </Hierarchy>
39      </Dimension>
40      <Dimension name="Distritos">
41          <Hierarchy hasAll="true" allMemberName="Todas las
            provincias" primaryKey="codigo_distrito">
42              <Table name="geografia_especial2"/>
43              <Level name="Provincia" type="String"
                column="nombre_provincia" uniqueMembers="false">
44                  <Property name="geom"
                    column="geometria_provincia" type="Geometry" />
45              </Level>
46              <Level name="Cantón" type="String"
                column="nombre_canton" uniqueMembers="false">
47                  <Property name="geom" column="geometria_canton"
                    type="Geometry" />
48              </Level>
49              <Level name="Distrito" type="String"
                column="nombre_distrito" uniqueMembers="false">
50                  <Property name="geom" column="geometria_distrito"
                    type="Geometry" />
```

```
51              </Level>
52          </Hierarchy>
53      </Dimension>
54      <Dimension name="Regiones CCP">
55          <Hierarchy hasAll="true" allMemberName="Todas las regiones
            del CCP" primaryKey="codigo_distrito">
56              <Table name="geografia_especial2"/>
57              <Level name="Región CCP" column="codigo_ccp"
                nameColumn="nombre_ccp" uniqueMembers="false">
58                  <Property name="geom" column="geometria_ccp"
                    type="Geometry" />
59              </Level>
60              <Level name="Distrito" column="codigo_distrito"
                nameColumn="nombre_distrito" uniqueMembers="false">
61                  <Property name="geom" column="geometria_distrito"
                    type="Geometry" />
62              </Level>
63          </Hierarchy>
64      </Dimension>
65      <Dimension name="Regiones CCSS">
66          <Hierarchy hasAll="true" allMemberName="Todas las regiones
            de la CCSS" primaryKey="codigo_distrito">
67              <Table name="geografia_especial2"/>
68              <Level name="Región de salud"
```

```
                column="codigo_region_ccss"
                nameColumn="nombre_region_ccss" uniqueMembers="false">
69                   <Property name="geom" dependsOnLevelValue="true"
                     column="geometria_region_ccss" type="Geometry"/>
70              </Level>
                <Level name="Área de salud" column="codigo_area_ccss"
71              nameColumn="nombre_area_ccss" uniqueMembers="false">
                     <Property name="geom" dependsOnLevelValue="true"
72                   column="geometria_area_ccss" type="Geometry"/>
73              </Level>
                <Level name="Distrito" column="codigo_distrito"
74              nameColumn="nombre_distrito" uniqueMembers="false">
                     <Property name="geom" dependsOnLevelValue="true"
75                   column="geometria_distrito" type="Geometry"/>
                </Level>
76
77          </Hierarchy>
78      </Dimension>
79      <Dimension name="Regiones INEC">
            <Hierarchy hasAll="true" allMemberName="Todas las regiones
80          del INEC" primaryKey="codigo_distrito">
81              <Table name="geografia_especial2"/>
                <Level name="Región INEC" column="codigo_inec"
82              nameColumn="nombre_inec" uniqueMembers="false">
                     <Property name="geom" column="geometria_inec"
83                   type="Geometry" />
```

```
84              </Level>
85              <Level     name="Distrito"     column="codigo_distrito"
                nameColumn="nombre_distrito" uniqueMembers="false">
86                   <Property name="geom" column="geometria_distrito"
                     type="Geometry" />
87              </Level>
88         </Hierarchy>
89    </Dimension>
90    <Dimension name="Subregiones INEC">
91         <Hierarchy     hasAll="true"     allMemberName="Todas     las
           subregiones del INEC" primaryKey="codigo_distrito">
92              <Table name="geografia_especial2"/>
93              <Level  name="Subregión  INEC"  column="codigo_sub_inec"
                nameColumn="nombre_sub_inec" uniqueMembers="false">
94                   <Property name="geom" column="geometria_sub_inec"
                     type="Geometry" />
95              </Level>
96              <Level     name="Distrito"     column="codigo_distrito"
                nameColumn="nombre_distrito" uniqueMembers="false">
97                   <Property name="geom" column="geometria_distrito"
                     type="Geometry" />
98              </Level>
```

```
 99          </Hierarchy>
100      </Dimension>
101      <Dimension name="GAM">
             <Hierarchy          hasAll="true"          allMemberName="Todo"
102          primaryKey="codigo_distrito">
103                  <Table name="geografia_especial2"/>
                     <Level          name="Area"          column="codigo_gam"
104                  nameColumn="nombre_gam" uniqueMembers="false">
                         <Property    name="geom"    column="geometria_gam"
105                      type="Geometry" />
106                  </Level>
                     <Level     name="Distrito"     column="codigo_distrito"
107                  nameColumn="nombre_distrito" uniqueMembers="false">
                         <Property name="geom" column="geometria_distrito"
108                      type="Geometry" />
109                  </Level>
110          </Hierarchy>
111      </Dimension>
112      <Dimension name="Zonas">
             <Hierarchy  hasAll="true"  allMemberName="Todas  las  zonas"
             primaryKey="codigo_distrito">
113
114                  <Table name="geografia_especial2"/>
                     <Level        name="Zona"        column="codigo_urbano"
                     nameColumn="nombre_urbano" uniqueMembers="false">
115
```

```
                    <Property name="geom" column="geometria_urbano"
116                 type="Geometry" />
117             </Level>
                <Level name="Distrito" column="codigo_distrito"
118             nameColumn="nombre_distrito" uniqueMembers="false">
                    <Property name="geom" column="geometria_distrito"
119                 type="Geometry" />
120             </Level>
121         </Hierarchy>
122     </Dimension>
123     <Cube name="poblacion">
124         <Table name="poblacion"/>
125         <Dimension name="Tiempo">
                <Hierarchy hasAll="true" allMemberName="Todos los
126             años" primaryKey="id">
                    <Level name="Año" type="Numeric" column="anno"
127                 uniqueMembers="false"/>
128             </Hierarchy>
129         </Dimension>
            <DimensionUsage name="Sexo" source="Sexo"
130         foreignKey="sexo"/>
            <DimensionUsage name="Edad" source="Edad"
131         foreignKey="edad"/>
            <DimensionUsage name="Distritos" source="Distritos"
132         foreignKey="distrito"/>
```

```
133         <DimensionUsage name="Regiones CCP" source="Regiones CCP"
            foreignKey="distrito"/>
134         <DimensionUsage name="Regiones CCSS" source="Regiones CCSS"
            foreignKey="distrito"/>
135         <DimensionUsage name="Regiones INEC" source="Regiones INEC"
            foreignKey="distrito"/>
136         <DimensionUsage name="Subregiones INEC" source="Subregiones
            INEC" foreignKey="distrito"/>
137         <DimensionUsage name="GAM" source="GAM"
            foreignKey="distrito"/>
138         <DimensionUsage name="Zonas" source="Zonas"
            foreignKey="distrito"/>
139         <Measure name="Cantidad de habitantes" column="poblacion"
            aggregator="sum" visible="false"/>
140     </Cube>
141     <Cube name="cancer">
142         <Table name="cancer_no_desconocidos"/>
143         <Dimension name="Tiempo">
144               <Hierarchy hasAll="true" allMemberName="Todos los
                  años" primaryKey="id">
145                     <Level name="Año" type="Numeric" column="anno"
                        uniqueMembers="false"/>
146               </Hierarchy>
147         </Dimension>
148         <DimensionUsage name="Sexo" source="Sexo"
```

```
        foreignKey="sexo"/>
        <DimensionUsage          name="Edad"          source="Edad"
149     foreignKey="edad"/>
        <DimensionUsage          name="Tipo"          source="Tipo"
150     foreignKey="localiz"/>
        <DimensionUsage     name="Distritos"     source="Distritos"
151     foreignKey="distrito"/>
        <DimensionUsage  name="Regiones  CCP"  source="Regiones  CCP"
152     foreignKey="distrito"/>
        <DimensionUsage name="Regiones CCSS" source="Regiones CCSS"
153     foreignKey="distrito"/>
        <DimensionUsage name="Regiones INEC" source="Regiones INEC"
154     foreignKey="distrito"/>
        <DimensionUsage name="Subregiones INEC" source="Subregiones
155     INEC" foreignKey="distrito"/>
        <DimensionUsage           name="GAM"           source="GAM"
156     foreignKey="distrito"/>
        <DimensionUsage         name="Zonas"         source="Zonas"
157     foreignKey="distrito"/>
158     <Measure name="Casos" column="casos" aggregator="sum"/>
        <Measure  name="Muertes"  column="muertes"  aggregator="sum"/>
159
160 </Cube>
161 <Cube name="defunciones">
162     <Table name="defunciones_no_desconocidos"/>
```

```
163         <Dimension name="Tiempo">
            <Hierarchy hasAll="true" allMemberName="Todos los
164         años" primaryKey="id">
                <Level name="Año" type="Numeric" column="anno"
165             uniqueMembers="false"/>
166         </Hierarchy>
167         </Dimension>
            <DimensionUsage name="Sexo" source="Sexo"
168         foreignKey="sexo"/>
            <DimensionUsage name="Edad" source="Edad"
169         foreignKey="edad"/>
            <DimensionUsage name="Causa" source="Causa"
170         foreignKey="causa"/>
            <DimensionUsage name="Distritos" source="Distritos"
171         foreignKey="distrito"/>
            <DimensionUsage name="Regiones CCP" source="Regiones CCP"
172         foreignKey="distrito"/>
            <DimensionUsage name="Regiones CCSS" source="Regiones CCSS"
173         foreignKey="distrito"/>
            <DimensionUsage name="Regiones INEC" source="Regiones INEC"
174         foreignKey="distrito"/>
            <DimensionUsage name="Subregiones INEC" source="Subregiones
175         INEC" foreignKey="distrito"/>
            <DimensionUsage name="GAM" source="GAM"
176         foreignKey="distrito"/>
```

```
177         <DimensionUsage name="Zonas" source="Zonas" foreignKey="distrito"/>
178         <Measure name="Muertes" column="muertes" aggregator="sum"/>
179     </Cube>
180     <Cube name="defunciones_infantiles">
181         <Table name="defunciones_infantiles_nueva"/>
182         <Dimension name="Tiempo">
183               <Hierarchy hasAll="true" allMemberName="Todos los años" primaryKey="id">
184                     <Level name="Año" type="Numeric" column="anno" uniqueMembers="false"/>
185               </Hierarchy>
186         </Dimension>
187         <Dimension name="Edad" foreignKey="edad">
188         <Hierarchy hasAll="true" allMemberName="Todas las edades" primaryKey="codigo">
189                     <Table name="horas"/>
190                     <Level name="Edad" column="codigo" nameColumn="nombre" uniqueMembers="false"/>
191               </Hierarchy>
192         </Dimension>
193         <DimensionUsage name="Sexo" source="Sexo" foreignKey="sexo"/>
```

```
194         <DimensionUsage name="Causa" source="Causa"
            foreignKey="causa"/>
195         <DimensionUsage name="Distritos" source="Distritos"
            foreignKey="distrito"/>
196         <DimensionUsage name="Regiones CCP" source="Regiones CCP"
            foreignKey="distrito"/>
197         <DimensionUsage name="Regiones CCSS" source="Regiones CCSS"
            foreignKey="distrito"/>
198         <DimensionUsage name="Regiones INEC" source="Regiones INEC"
            foreignKey="distrito"/>
199         <DimensionUsage name="Subregiones INEC" source="Subregiones
            INEC" foreignKey="distrito"/>
200         <DimensionUsage name="GAM" source="GAM"
            foreignKey="distrito"/>
201         <DimensionUsage name="Zonas" source="Zonas"
            foreignKey="distrito"/>
202         <Measure name="Muertes" column="muertes" aggregator="sum"/>
203     </Cube>
204     <Cube name="nacimientos">
205         <Table name="nacimientos"/>
206         <Dimension name="Tiempo">
207               <Hierarchy hasAll="true" allMemberName="Todos los
                  años" primaryKey="id">
208                     <Level name="Año" type="Numeric" column="anno"
                        uniqueMembers="false"/>
```

```
209                 </Hierarchy>
210         </Dimension>
            <DimensionUsage          name="Sexo"          source="Sexo"
211         foreignKey="sexo"/>
            <DimensionUsage     name="Distritos"     source="Distritos"
212         foreignKey="distrito"/>
            <DimensionUsage  name="Regiones  CCP"  source="Regiones  CCP"
213         foreignKey="distrito"/>
            <DimensionUsage name="Regiones CCSS" source="Regiones CCSS"
214         foreignKey="distrito"/>
            <DimensionUsage name="Regiones INEC" source="Regiones INEC"
215         foreignKey="distrito"/>
            <DimensionUsage name="Subregiones INEC" source="Subregiones
216         INEC" foreignKey="distrito"/>
            <DimensionUsage           name="GAM"           source="GAM"
217         foreignKey="distrito"/>
            <DimensionUsage         name="Zonas"         source="Zonas"
218         foreignKey="distrito"/>
            <Measure       name="Nacimientos"       column="nacimientos"
219         aggregator="sum"/>
220     </Cube>
221     <VirtualCube name="defunciones_infantiles_nacimientos">
222         <CubeUsages>
                    <CubeUsage           cubeName="defunciones_infantiles"
223                 ignoreUnrelatedDimensions="true"/>
```

```
224             <CubeUsage cubeName="nacimientos"/>
225         </CubeUsages>
226         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Causa"/>
227         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Edad"/>
228         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Tiempo"/>
229         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Sexo"/>
230         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Distritos"/>
231         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Regiones CCP"/>
232         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Regiones CCSS"/>
233         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Regiones INEC"/>
234         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Subregiones INEC"/>
235         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="GAM"/>
236         <VirtualCubeDimension      cubeName="defunciones_infantiles"
            name="Zonas"/>
237         <VirtualCubeMeasure        cubeName="defunciones_infantiles"
```

```
        name="[Measures].[Muertes]"/>
        <VirtualCubeMeasure cubeName="nacimientos"
238     name="[Measures].[Nacimientos]"/>
        <CalculatedMember name="Tasa de mortalidad infantil por
239     1000" dimension="Measures">
              <Formula>Iif((ISEMPTY(Aggregate(Crossjoin({[Causa_infa
              ntil].[Todas las causas]},{[Edad].[Todas las
              edades]}), [Measures].[Nacimientos])) OR
              (Aggregate(Crossjoin({[Causa_infantil].[Todas las
              causas]},{[Edad].[Todas las edades]}),
              [Measures].[Nacimientos])) = 0), NULL,
              ([Measures].[Muertes]/(Aggregate(Crossjoin({[Causa_inf
              antil].[Todas las causas]},{[Edad].[Todas las
240           edades]}), [Measures].[Nacimientos]))*1000))</Formula>
241     </CalculatedMember>
    </VirtualCube>
242
243 <VirtualCube name="nacimientos_poblacion">
244     <CubeUsages>
              <CubeUsage cubeName="nacimientos"
245           ignoreUnrelatedDimensions="true"/>
246           CubeUsage cubeName="poblacion"/>
247     </CubeUsages>
248     <VirtualCubeDimension cubeName="nacimientos" name="Tiempo"/>
249     <VirtualCubeDimension cubeName="nacimientos" name="Sexo"/>
250     <VirtualCubeDimension cubeName="nacimientos"
```

```
        name="Distritos"/>
        <VirtualCubeDimension cubeName="nacimientos" name="Regiones
251     CCP"/>
        <VirtualCubeDimension cubeName="nacimientos" name="Regiones
252     CCSS"/>
        <VirtualCubeDimension cubeName="nacimientos" name="Regiones
253     INEC"/>
        <VirtualCubeDimension                cubeName="nacimientos"
254     name="Subregiones INEC"/>
255     <VirtualCubeDimension cubeName="nacimientos" name="GAM"/>
256     <VirtualCubeDimension cubeName="nacimientos" name="Zonas"/>
        <VirtualCubeMeasure                    cubeName="poblacion"
257     name="[Measures].[Cantidad de habitantes]"/>
        <VirtualCubeMeasure                  cubeName="nacimientos"
258     name="[Measures].[Nacimientos]"/>
        <CalculatedMember name="Tasa de natalidad por 1000"
259     dimension="Measures">
              <Formula>Iif((ISEMPTY([Measures].[Cantidad de
              habitantes]) OR [Measures].[Cantidad de habitantes] =
              0),                                          NULL,
              (([Measures].[Nacimientos]/[Measures].[Cantidad de
260           habitantes])*1000))</Formula>
261     </CalculatedMember>
    </VirtualCube>
262
```

```
263     <VirtualCube name="cancer_poblacion">
264         <CubeUsages>
265             <CubeUsage cubeName="cancer"
                ignoreUnrelatedDimensions="true"/>
266             CubeUsage cubeName="poblacion"/>
267         </CubeUsages>
268         <VirtualCubeDimension cubeName="cancer" name="Edad"/>
269         <VirtualCubeDimension cubeName="cancer" name="Tiempo"/>
270         <VirtualCubeDimension cubeName="cancer" name="Sexo"/>
271         <VirtualCubeDimension cubeName="cancer" name="Tipo"/>
272         <VirtualCubeDimension cubeName="cancer" name="Distritos"/>
273         <VirtualCubeDimension cubeName="cancer" name="Regiones
            CCP"/>
274         <VirtualCubeDimension cubeName="cancer" name="Regiones
            CCSS"/>
275         <VirtualCubeDimension cubeName="cancer" name="Regiones
            INEC"/>
276         <VirtualCubeDimension cubeName="cancer" name="Subregiones
            INEC"/>
```

```
277         <VirtualCubeDimension cubeName="cancer" name="GAM"/>
278         <VirtualCubeDimension cubeName="cancer" name="Zonas"/>
279         <VirtualCubeMeasure cubeName="poblacion"
            name="[Measures].[Cantidad de habitantes]"/>
280         <VirtualCubeMeasure cubeName="cancer"
            name="[Measures].[Casos]"/>
281         <VirtualCubeMeasure cubeName="cancer"
            name="[Measures].[Muertes]"/>
282         <CalculatedMember name="Tasa de incidencia (por 100000)"
            dimension="Measures">
283               <Formula>Iif((ISEMPTY(Aggregate({[Tipo].[Todos los
                  tipos]}, [Measures].[Cantidad de habitantes])) OR
                  (Aggregate({[Tipo].[Todos los tipos]},
                  [Measures].[Cantidad de habitantes])) = 0), NULL,
                  ([Measures].[Casos]/(Aggregate({[Tipo].[Todos los
                  tipos]}, [Measures].[Cantidad de
                  habitantes]))*100000))</Formula>
284         </CalculatedMember>
285         <CalculatedMember name="Tasa de mortalidad (por 10000)"
            dimension="Measures">
286               <Formula>Iif((ISEMPTY(Aggregate({[Tipo].[Todos los
```

```
                tipos]}, [Measures].[Cantidad de habitantes])) OR
                (Aggregate({[Tipo].[Todos los tipos]},
                [Measures].[Cantidad de habitantes])) = 0), NULL,
                ([Measures].[Muertes]/(Aggregate({[Tipo].[Todos los
                tipos]}, [Measures].[Cantidad de
                habitantes]))*10000))</Formula>
287         </CalculatedMember>
288     </VirtualCube>
289     <VirtualCube name="defunciones_poblacion">
290         <CubeUsages>
291             <CubeUsage cubeName="defunciones"
                ignoreUnrelatedDimensions="true"/>
292             <CubeUsage cubeName="poblacion"/>
293         </CubeUsages>
294         <VirtualCubeDimension cubeName="defunciones" name="Edad"/>
295         <VirtualCubeDimension cubeName="defunciones" name="Tiempo"/>
296         <VirtualCubeDimension cubeName="defunciones" name="Sexo"/>
297         <VirtualCubeDimension cubeName="defunciones" name="Causa"/>
298         <VirtualCubeDimension cubeName="defunciones"
            name="Distritos"/>
```

```
299         <VirtualCubeDimension cubeName="defunciones" name="Regiones
            CCP"/>
300         <VirtualCubeDimension cubeName="defunciones" name="Regiones
            CCSS"/>
301         <VirtualCubeDimension cubeName="defunciones" name="Regiones
            INEC"/>
302         <VirtualCubeDimension cubeName="defunciones"
            name="Subregiones INEC"/>
303         <VirtualCubeDimension cubeName="defunciones" name="GAM"/>
304         <VirtualCubeDimension cubeName="defunciones" name="Zonas"/>
305         <VirtualCubeMeasure cubeName="poblacion"
            name="[Measures].[Cantidad de habitantes]"/>
306         <VirtualCubeMeasure cubeName="defunciones"
            name="[Measures].[Muertes]"/>
307         <CalculatedMember name="Tasa de mortalidad (por 10000)"
            dimension="Measures">
308               <Formula>Iif((ISEMPTY(Aggregate({[Causa].[Todas las
                  causas]}, [Measures].[Cantidad de habitantes])) OR
                  (Aggregate({[Causa].[Todas las causas]},
                  [Measures].[Cantidad de habitantes])) = 0), NULL,
                  ([Measures].[Muertes]/(Aggregate({[Causa].[Todas las
```

```
                causas]},            [Measures].[Cantidad            de
                habitantes]))*10000))</Formula>
        </CalculatedMember>
    </VirtualCube>
</Schema>
```

ANEXO I – ARTÍCULO PRESENTADO EN CLEI 2013

Spatial Data Warehouses and SOLAP Using Open-Source Tools

Diana Bogantes González
Escuela de Ciencias de Computación e Informática
Universidad de Costa Rica
Alajuela, Costa Rica
diana.bogantesgonzalez@ucr.ac.cr

Leonardo Pandolfi González
Escuela de Ciencias de Computación e Informática
Universidad de Costa Rica
San José, Costa Rica
leonardo.pandolfi@ucr.ac.cr

***Abstract*—The use of data warehouses and OLAP tools has been increasing in the last few decades. Several organizations are seeking to extend those concepts in order to include spatial elements within the data analysis. However, having one tool that includes both OLAP operations and results presentation on geographic maps has proven to be challenging. If we add the requirement of using open-source software to achieve this, the scenario becomes even more defiant. This article uses a practical example to expose different challenges that can arise when working with spatial data warehouses and spatial OLAP. It also describes possible solutions offered by open-source tools in order to implement this type of applications.**

Keywords—spatial OLAP; spatial data warehouse; virtual cubes; calculated measures

I. INTRODUCCIÓN

Es natural que un negocio implique la acumulación de datos a lo largo del tiempo; sin embargo, esto también causa que un gran número de instituciones y compañías alrededor del mundo experimenten a diario la necesidad de tener un proceso de observación de datos cada vez más robusto. La utilización de sistemas computacionales para resolver tal complicación hizo posible que los datos históricos se convirtieran en piezas valiosas del análisis y por ende, los transformó en determinantes poderosos para la toma de decisiones.

Debido a la constante búsqueda de nuevas opciones para el estudio de los datos, se han incorporado en él disciplinas que van más allá del área de la computación. Un campo todavía nuevo dentro de este fenómeno es el que involucra al componente espacial como un factor anexo en el análisis. Grandes empresas como Microsoft, Oracle e IBM, han incorporado el elemento espacial a sus motores de bases y almacenes de datos. Igualmente, sistemas de gestión *open-source* como MySQL y PostgreSQL han lanzado extensiones para el manejo de datos espaciales. Opciones de almacenamiento como las anteriores, combinadas con herramientas de procesamiento analítico en línea espacial (SOLAP – *Spatial Online Analytical Processing*) enriquecen el análisis al facilitar la visualización de resultados y ofrecen así el mejor soporte para la toma de decisiones.

La relevancia que se le ha dado al uso de elementos espaciales en el análisis de datos radica en que más del 80% de datos de los negocios tienen algún tipo de contexto espacial asociado, generalmente en forma de código postal y direcciones [5,20]. Al utilizar herramientas que permiten integrar estos datos dentro del análisis por medio de su representación espacial, es posible revelar relaciones entre los datos que de otra forma son difíciles de descubrir [20].

En sus inicios, el análisis de datos espaciales se realizaba mediante Sistemas de Información Geográfica (GIS – *Geographic Information Systems*),

los cuales evolucionaron a partir de los campos de geografía, cartografía y bases de datos. Con asiduidad, las aplicaciones GIS requieren que el usuario posea o adquiera conocimiento en estos campos para poder utilizarlas [4]. Adicionalmente, los vendedores se han enfocado en la funcionalidad y no en la facilidad de uso de las interfaces; razones por las cuales usuarios no expertos en estos campos consideran este tipo de aplicaciones como inviables para ser utilizadas en el análisis [4].

Por su parte, el uso de SOLAP como alternativa a los GIS permite a usuarios no expertos analizar información en una forma más sencilla e intuitiva, mediante la formulación de consultas a través de una interfaz gráfica que les permite combinar parámetros según lo necesiten. En general, las herramientas SOLAP permiten a los usuarios seleccionar fácil y rápidamente los parámetros de las consultas, así como visualizar los resultados en tablas dinámicas y mapas. Esto permite que expertos en otras áreas, como la estadística, encuentren utilidad en las herramientas SOLAP a modo de medio para analizar información y encontrar patrones.

En Costa Rica, la situación no difiere. Instituciones como el Centro Centroamericano de Población (CCP) y el Instituto Nacional de Estadística y Censos (INEC) buscan aprovechar los datos demográficos y espaciales disponibles para descubrir patrones con respecto a la natalidad, mortalidad e incidencia de cáncer en la población costarricense, de previo a la aplicación de herramientas de estadística tradicionales.

El presente artículo se enfocará en describir los componentes propios de almacenes de datos espaciales y SOLAP. Para ello, presentará un caso de estudio que surgió de la necesidad planteada por el CCP y mostrará la implementación del sistema de análisis de datos a través de algunas soluciones *open-source* disponibles en la actualidad. Además, se describirán diferentes aspectos, unos ventajosos y otros desafiantes, que pueden presentarse durante el diseño y la implementación, explicando a la vez las posibilidades de ajustar el *software* actual para resolver necesidades de análisis particulares.

II. TRABAJOS RELACIONADOS

El interés en el análisis de información espacial tiene una larga historia. Los avances individuales en las áreas de almacenes de datos y GIS lograron impulsar y complementar ambos campos para desarrollar mejores aplicaciones que, mediante el uso de datos espaciales, son utilizadas para el descubrimiento de conocimiento. No obstante, con el paso de los años, los requerimientos en el área de geomática han expuesto numerosos problemas al combinar tecnologías espaciales y no-espaciales, por ejemplo, tiempos de respuesta lentos, gran tamaño de las tablas y pocas opciones de navegación a partir de los mapas [20].

Específicamente relacionado con SOLAP, se recalca que el aspecto de visualización es el más significativo y a la vez el más desafiante dentro de las herramientas de este tipo [16]. La interfaz debería permitir combinación de dimensiones espaciales, navegación entre niveles cartográficos y textuales, y por último, soportar el análisis multidimensional utilizando mapas, tablas y gráficos en forma sincronizada [16,17].

Tomando en cuenta estos requerimientos, se han creado diferentes herramientas SOLAP; sin embargo, la mayoría de ellas son herramientas propietarias. *JMap Spatial OLAP*, ahora llamada *Map4Decision* es pregonada como la primera tecnología web que integró bases de datos espaciales dentro de un ambiente de soporte de decisiones [21]. *GeWOlap* está basada en una arquitectura de tres capas: (1) sistema de gestión de base de datos objeto-relacional, (2) servidor OLAP y (3) capa cliente que combina OLAP y GIS, implementada utilizando JPivot [16]. Asimismo, se han desarrollado herramientas hechas a la medida para un área de análisis específica; la propuesta por Scotch y Parmanto en [13] combina OLAP con GIS a fin de analizar información de la salud pública. Además, equipos de investigación como los mencionados en [10, 21] han propuesto alternativas para modelado y consulta de datos espaciales.

En el ambiente de *software* libre, se propone una herramienta SOLAP que utiliza servicios web [10]. Por otra parte, la compañía *SpagoBI Competency Center* ofrece una suite de herramientas para inteligencia de negocios (BI – *Business Intelligence*) [16]. Basadas en más de treinta motores analíticos desarrollados por la empresa, *SpagoBI* brinda diferentes opciones para realizar manipulación y análisis de información. En cuanto al ámbito geográfico, la herramienta ofrece operaciones propias de GIS, como el cálculo de áreas y distancias; permite además representar medidas en las divisiones territoriales mostradas en el mapa a manera de un mapa de calor (*heatmap*) basado en una escala de colores. Sin embargo, aunque tiene la capacidad para presentar datos agregados mediante la selección de diferentes capas, no contiene

funcionalidad para moverse por los niveles mediante acciones directas sobre el mapa; por ejemplo, no es posible desplegar las subregiones al hacer clic en una región.

A pesar de la variedad de opciones ofrecida por la suite de *SpagoBI*, esta cuenta con la limitante de que sus componentes de visualización espacial y operaciones OLAP se encuentran separados [15]. En el ambiente de *software* libre no existe al día de hoy una herramienta SOLAP de uso general que posea las capacidades y funcionalidades necesarias para un ambiente de soporte de decisiones eficiente.

III. MODELO MULTIDIMENSIONAL ESPACIAL

El ciclo de vida del proceso de diseño de un almacén de datos espacial inicia con el diseño y la planificación, fases en las que se define el ámbito del proyecto y los requerimientos del negocio [5]. Una de las etapas consecuentes es la creación de un esquema conceptual según la especificación de requerimientos de la fase previa. Este esquema puede ser originado a partir del uso de modelos como *MultiDim* [5], que permite representar elementos espaciales y no espaciales en el mismo diagrama.

MultiDim toma como base el modelo multidimensional, que se basa en una visión abstracta llamada **cubo**. Los cubos hacen referencia a los focos de análisis que se definieron como parte de los requerimientos; por ejemplo en la Figura 2, los cubos *Cáncer* y *Población* (representados por rombos grises) se refieren a la necesidad de analizar datos sobre la población y los diferentes tipos de cáncer desde diferentes perspectivas, como rango de edad, género y ubicación geográfica. En general, un cubo está formado por [5]:

- **Medidas**: hechos del negocio que se desean analizar, como por ejemplo: (1) *Casos*, (2) *Muertes* y (3) *Tasa de incidencia* para el cubo *Cáncer*.
- **Dimensiones**: conjunto de atributos que sirven para identificar y categorizar medidas desde distintas perspectivas. Por ejemplo, en la Figura 2, *Tiempo*, *Edad*, *Tipo Cáncer*, *Sexo* son algunas de las dimensiones del cubo *Cáncer*.
- **Miembros**: las instancias de una dimensión. Por ejemplo, *Cáncer de Estómago* y *Leucemia* son miembros de la dimensión *Tipo Cáncer* del cubo *Cáncer*.
- **Jerarquías**: son los niveles de detalle que presentan los miembros de la dimensión. Las jerarquías pueden tener distinto número de **niveles**. Por ejemplo, en la Figura 2 se puede observar una jerarquía con los niveles de *Distrito* → *Cantón* → *Provincia* y otra jerarquía paralela de *Distrito* → *Área de Salud* → *Región de Salud*. Para distinguir una jerarquía en el esquema, se utiliza un óvalo con su nombre y se coloca sobre la representación de su nivel más bajo. Por ejemplo, en la Figura 2, las jerarquías llamadas *Ubicación, INEC* y *CCSS* son originadas desde el nivel de *Distrito*.

Adicionalmente a los elementos mencionados, un almacén de datos espacial incorpora objetos espaciales y permite realizar operaciones sobre ellos. Un objeto espacial corresponde a una entidad del mundo real representada a través de un componente descriptivo y un componente espacial [5]. El componente descriptivo incluye las características del objeto, representándolo mediante tipos de datos convencionales como *String*, *int* y *date*. A manera de ejemplo, un objeto *Provincia* puede ser descrito a través de un nombre y un código. Por su parte, el componente espacial se caracteriza por contener la geometría. Una **geometría** es un conjunto de puntos descritos a través de coordenadas de latitud y longitud. La Figura 1 incluye los principales tipos de geometrías y su uso común:

- **Punto**: denota una locación singular en el espacio, por ejemplo, para representar la localización de una tienda.
- **Multipunto**: conjunto de puntos, por ejemplo, para indicar la localización de casas en un pueblo.
- **Línea**: geometría de una dimensión que corresponde a un conjunto de puntos conectados que definen segmentos de recta o curva. Ésta se puede utilizar para representar, por ejemplo, una carretera.
- **Multilínea**: agrupación de líneas que sirve, por ejemplo, para ubicar un sistema de carreteras.
- **Polígono**: geometría de dos dimensiones que denota un conjunto de puntos conectados, formando una superficie. Se puede usar para representar provincias, cantones y distritos.
- **Multipolígono**: conjunto de polígonos. Se utiliza para representar regiones cuya

geometría está compuesta por más de un polígono; por ejemplo, un distrito formado por islas.

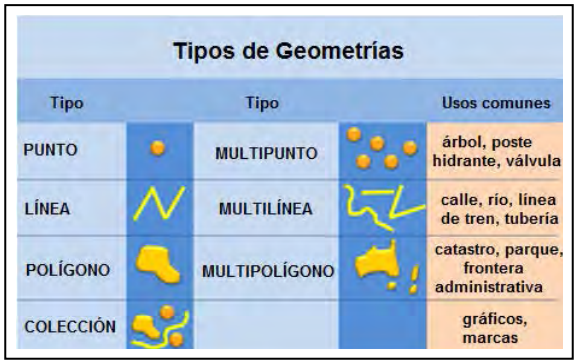


Fig. 1. Tipos de geometrías [8]

Al igual que un almacén de datos convencional, un almacén de datos espacial contiene elementos que representan los enfoques de análisis, medidas, dimensiones, jerarquías y niveles. La diferencia radica en que se le incorpora el componente espacial, creando así:

- **Medidas espaciales**: representadas por una geometría, como la ubicación de un accidente.
- **Dimensiones espaciales**: contienen por lo menos una jerarquía espacial.
- **Jerarquías espaciales**: contienen niveles espaciales. Por ejemplo: distrito, cantón y provincia, cada uno asociado con su geometría, como se indica en la Figura 2 por medio del pictograma a la par del nombre del nivel. En el caso de los **niveles espaciales**, sus miembros están representados por medio de una geometría; por ejemplo, una provincia puede ser representada por un polígono o un multipolígono.

Un almacén de datos espacial que opere en conjunto con herramientas SOLAP permite efectuar análisis diversos sobre los datos disponibles. Basado en el modelo multidimensional, SOLAP permite crear consultas complejas de una forma gráfica, combinando medidas y dimensiones según se necesite. El usuario puede elegir la forma en que los resultados son visualizados a través de tablas, gráficos y mapas. Asimismo, es posible crear nuevas consultas a partir de los resultados de una consulta previa; por ejemplo, mediante la modificación de filas y columnas.

La utilización de cubos (S)OLAP posibilita la ejecución de distintas operaciones durante el análisis, por ejemplo:

- **Roll-up**: permite subir en la escala de niveles de la jerarquía. La operación agrupa los miembros de un mismo nivel en clasificaciones más generales correspondientes a un nivel superior y, en forma simultánea, realiza la agregación de medidas. Por ejemplo, considerando la jerarquía *Distrito* → *Cantón* → *Provincia* de la Figura 2 junto con la medida *Casos* del cubo *Cáncer* y aplicándole *roll-up* desde *Distrito* hacia *Cantón*, se suman los casos de cáncer correspondientes a los distritos de cada cantón; por consiguiente, se obtiene el total de casos distribuido por cantones.

  Drill-down: operación opuesta a *roll-up* que se utiliza para navegar desde el nivel superior (el más general) hacia el nivel inferior (el más detallado) de la jerarquía. Usando el esquema de la Figura 2, al hacer *drill-down* desde *Región Salud* hacia el nivel de *Área Salud* sobre la medida *Cantidad de habitantes* del cubo *Cáncer*, se obtiene como resultado la cantidad de habitantes distribuida entre cada una de las áreas de salud.

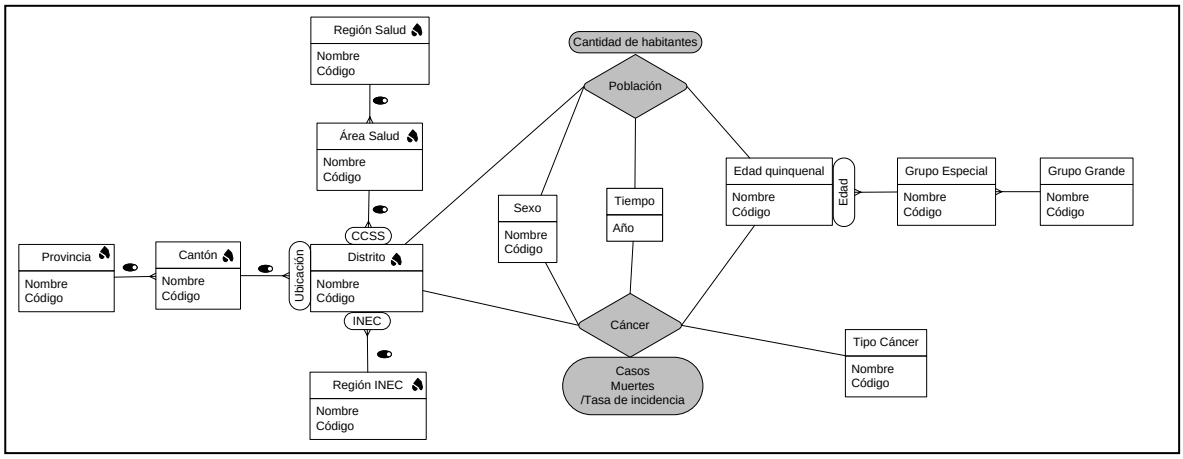


Fig. 2. Modelo MultiDim para la aplicación del CCP

- **Slice-and-dice**: ***slice*** consiste en obtener un subconjunto de datos mediante la selección de algunos miembros de una dimensión. Por ejemplo, al seleccionar el miembro *San José* del nivel *Provincias* en el cubo *Cáncer*, se obtiene el número de casos de cáncer únicamente para la provincia escogida. Cuando esta operación se realiza en más de dos dimensiones del cubo de datos, se utiliza el término ***dice***.
- **Pivot**: rota los ejes del cubo, permitiendo mostrar distintas presentaciones del mismo conjunto de datos. Por ejemplo, si originalmente se presentan casos de cáncer con los elementos de *Tipo Cáncer* representados por filas y cada una de las *Provincias* desplegadas en columnas, la ejecución de *pivot* posiciona ese mismo conjunto de resultados con *Provincias* en las filas y *Tipo Cáncer* en las columnas.

IV. CASO PRÁCTICO

En Costa Rica, el CCP tiene a su disposición datos sobre nacimientos, defunciones e incidencia de diferentes tipos de cáncer recopilados desde los años ochenta. El Centro pone estos datos a disposición del público mediante su sitio web, que contiene accesos directos a diferentes bases de datos. La información es accesible únicamente en forma separada para cada base de datos y las opciones de consulta son limitadas. Además, la selección de variables y operadores requiere de cierto conocimiento previo del sistema; si no se cuenta con él, es probable que se recurra frecuentemente a la sección de ayuda para aprender la manera exacta como deben ser formuladas las consultas.

Por estas razones, la directiva del CCP planteó la necesidad de unificar datos de diferentes fuentes en un solo sistema que permitiera visualizarlos por medio de tablas, gráficos y mapas. La idea fundamental era hacer posible la consecución de un análisis exploratorio a partir de los datos, de modo que se pudieran detectar patrones y ayudar en la toma de decisiones de interés local o nacional sin requerir conocimientos especializados en estadística, cartografía o minería de datos. Para resolver los requerimientos anteriores, se planteó el uso de almacenes de datos y OLAP; asimismo, se diseñó un esquema conceptual que aborda las diferentes necesidades de análisis.

La Figura 2 muestra un extracto del esquema conceptual desarrollado para este caso particular. Utilizando notaciones del modelo *MultiDim* [5], se presentan dos cubos (uno por cada foco de análisis): *Cáncer* y *Población*. Ambos comparten las dimensiones originadas a partir de *Sexo*, *Tiempo*, *Edad quinquenal* y *Distrito*, sin embargo *Cáncer* posee un mayor nivel de granularidad. Esto significa que, adicionalmente a las categorías mencionadas, los datos de ese cubo pueden descomponerse también por *Tipo Cáncer*, mientras que los de *Población* no. En la Figura 2 también se puede apreciar que *Distrito* da origen a las jerarquías *Ubicación*, *CCSS* e *INEC*. Las tres aludidas constituyen representaciones de áreas geográficas, lo que las hace propicias para ser utilizadas en

dimensiones espaciales. En cuanto a las medidas, *Población* contiene sólo *Cantidad de habitantes*, que refleja la cantidad de habitantes, mientras que *Cáncer* contiene los casos de cáncer (*Casos*), las muertes causadas por cáncer (*Muertes*) y la tasa de incidencia de cáncer (*Tasa de incidencia*).

V. ALMACENES DE DATOS ESPACIALES EN POSTGIS

El sistema propuesto en el esquema conceptual se materializó a través de utilización de *software* libre. Esto en congruencia con el impulso que le ha dado el gobierno costarricense al uso de este tipo de herramientas, mismo que se ha hecho manifiesto al darle prioridad sobre el *software* propietario en las instituciones públicas [2].

Los datos que conforman cada cubo se obtienen a partir de un almacén de datos; para implementarlo en este caso particular se usó PostgreSQL, junto con su extensión espacial PostGIS. Por su parte, los objetos espaciales originales estaban en *shapefiles*, cuyo contenido es importado al almacén de datos espacial. *Shapefile* es un formato de datos desarrollado por el Instituto de Investigación en Sistemas Ambientales (*Environmental Systems Research Institute* - ESRI), que surgió como una opción para representar geometrías de fácil edición y trazado [6]. Debido a sus beneficios, como un menor gasto de espacio en disco y una mayor velocidad de trazado, estos archivos son utilizados en gran medida por aplicaciones que involucran el trazado de mapas. Diversos sistemas de administración de bases de datos como Oracle y PostgreSQL/PostGIS incluyen funciones para importarlos.

La selección de PostgreSQL y PostGIS se dio al considerar sus facilidades para el manejo de datos espaciales, entre ellas [1]:

- Soporte para importar y exportar *shapefiles* mediante línea de comandos o interfaz gráfica.
- Soporte de interoperabilidad abierta por medio de importación, exportación y representación de los datos en formatos como KML (*Keyhole Mark-up Language*), GML (*Geographic Mark-up Language*) y WKT (*Well-Known Text*).
- Soporte de amplia gama de operaciones espaciales, tanto para la creación de geometrías como para la manipulación y análisis de los datos espaciales.

PostGIS almacena la información geográfica en columnas de tipo *geometry*. A las mismas se les debe definir el identificador de referencia espacial (SRID – *Spatial Reference Identifier*) en el momento de su creación. La importancia de seleccionar el SRID correcto radica en el hecho de que contiene todos los metadatos sobre el sistema de coordenadas que se usa al colectar los datos. Esto hace posible el mapeado correcto durante el dibujo de los mapas y los cálculos acertados en operaciones sobre las geometrías, como área y perímetro. En el proyecto se utilizó el SRID 4326, que corresponde al WGS84 (Sistema Geodésico Mundial 1984 - *World Geodetic System* 84), ampliamente usado a nivel mundial.

A partir de PostGIS 2.0, dentro de la sentencia de creación de la tabla se especifican las columnas de tipo *geometry*. Por ejemplo, para crear la tabla *cantones* con su respectiva columna de tipo *geometry* se utilizó la operación presentada en la Figura 3.

```
CREATE TABLE cantones (
codigo_canton       int4,
nombre_canton       varchar(50),
codigo_provincia    int4
geometria_canton    geometry(POLYGON,4326) )
```

Fig. 3. Sentencia para agregar geometría a tabla existente

PostGIS permite rastrear y reportar los tipos de geometrías usados en la base de datos por medio de [14]:

- Spatial_ref_sys: tabla que contiene todos los sistemas de referencia espacial que pueden ser utilizados en la base de datos.
- Geometry_colums: vista que contiene la lista de todas las propiedades asociadas con columnas de tipo *geometry*.

La inserción de datos espaciales se puede realizar de distintas maneras. En caso de que la definición de las geometrías se encuentre en formatos como KML, WKT y GML, es posible utilizar funciones predefinidas en PostGIS para hacer su importación. Por ejemplo, la Figura 4 muestra la inserción de una parte de geometrías que representan a un cantón en la base de datos. El dato está en formato KML y se utiliza la función ST_GeomFromKML de PostGIS para importarla como geometría en la tabla *cantones*.

```
INSERT INTO cantones (geometria_canton)
VALUES ((Select ST_GeomFromKML
('<Polygon><outerBoundaryIs><LinearRing><coordinates>
-84.089277,9.947233 -84.083204,9.946468 -84.080159,9.94496
-84.080161,9.942705 -84.066482,9.942694 -84.065568,9.94089
...
</coordinates></LinearRing></outerBoundaryIs></Polygon>')))
```

Fig. 4. Inserción de geometría a través de PostGIS

VI. CARACTERÍSTICAS DE GEOMONDRIAN

Como parte de la arquitectura típica de un almacén de datos, es necesario implementar las capas de servidor OLAP y *front-end* para permitir la creación de cubos OLAP, así como la manipulación y visualización de los datos [5]. De acuerdo a la investigación realizada, sólo se encontró un servidor OLAP *open-source* que estuviera disponible públicamente y tuviera la capacidad de manejar datos espaciales: GeoMondrian. Su selección se fundamentó en los requerimientos inherentes al proyecto, en relación con la necesidad de incorporar representaciones de espacios geográficos. GeoMondrian es una herramienta creada con el objetivo de posibilitar el análisis espacial en Mondrian, el servidor OLAP de Pentaho. A las opciones básicas de Mondrian, GeoMondrian añade el manejo de datos espaciales y funciones específicas para ejecutar sobre ellos. Como fue desarrollado a partir de Mondrian, GeoMondrian heredó su funcionalidad básica y estructura; parte de esa estructura es la manera en que los cubos de datos son consultados.

A. *Uso de MDX e intefaz gráfica para formulación de consultas*

GeoMondrian emplea el lenguaje MDX (*Multi-Dimensional eXpressions*) para realizar consultas sobre el almacén de datos y así poder materializar las estructuras definidas en el esquema XML. Una consulta en MDX contiene la siguiente información [3]: (1) los miembros de los ejes que van a ser desplegados, (2) el nombre del cubo sobre el cual se hace la consulta y (3) el miembro que se utiliza para delimitar los datos retornados por la consulta (en caso de que se quiera aplicar *slice-and-dice*).

En la Figura 5, se muestra una consulta en MDX, donde los elementos de la dimensión *Distritos* y las medidas *Casos* y *Área en Km2* son los miembros de los ejes para desplegar. *Cáncer* es el cubo que se va a utilizar en la consulta y el elemento *2005* de la dimensión de *Tiempo* es utilizado para filtrar los resultados por ese año específico (*slice-and-dice*). Los alias *ON COLUMNS* y *ON ROWS* sirven para especificar qué valores van en los ejes X y Y de la tabla de resultados.

GeoMondrian utiliza JPivot (integrado en Mondrian) [11] para el despliegue de los datos, un visualizador que permite realizar las operaciones OLAP básicas, así como presentar los resultados en tablas y gráficos. Además, proporciona opciones gráficas para crear consultas a través la especificación de medidas y dimensiones. En dado caso, el código en MDX asociado a la consulta se genera automáticamente y es accesible desde la interfaz a través del editor de MDX, tal y como se muestra en la parte superior de la Figura 5. Si se modifica la consulta, por ejemplo, haciendo *drill-down*, el código MDX se actualiza automáticamente.

La posibilidad de ejecutar operaciones sobre datos espaciales constituye una de las principales características de GeoMondrian. Como un medio para plasmarla en el sistema, la herramienta añade funciones dedicadas al análisis geoespacial dentro del lenguaje MDX. El resultado de ese trabajo conjunto entre MDX y datos espaciales es llamado GeoMDX. Las funciones espaciales incluidas aportan alternativas para realizar operaciones sobre las geometrías, como lo son el cálculo de intersecciones, uniones, áreas, centroides y distancias.

En la Figura 5, la definición de la medida *Área en Km2* implica el uso de dos funciones espaciales; *ST_Transform* convierte las geometrías de un sistema de coordenadas basado en longitud y latitud (código 4326) a uno que utiliza kilómetros como unidad (código 32617), mientras que *ST_Area* calcula el área de los elementos de la dimensión *Distritos*. Los resultados para cada provincia se muestran en la parte inferior, tanto en tabla como en gráfico.

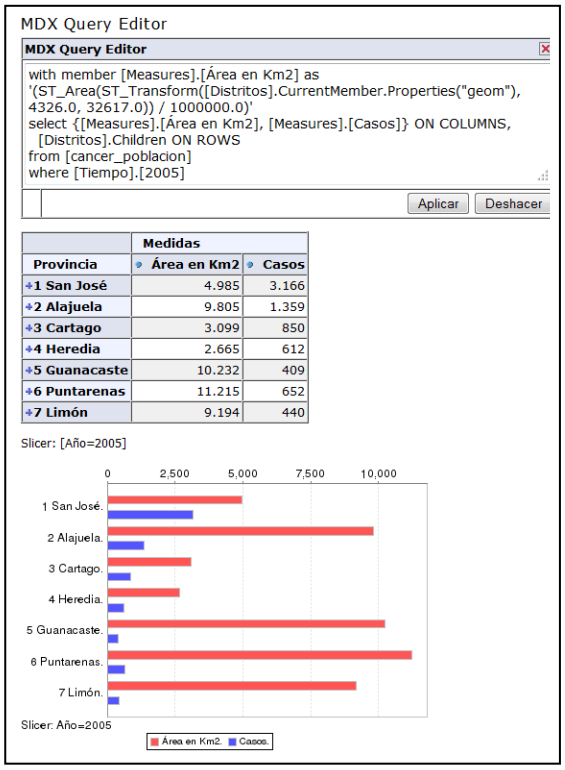


Fig. 5. Consulta GeoMDX y su resultado gráfico en GeoMondrian

B. *Diseño básico del esquema*

GeoMondrian construye cubos a partir de esquemas definidos por medio de archivos en formato XML. En ellos se especifican los elementos básicos que debe incluir el análisis, como las dimensiones, medidas y jerarquías. La Figura 6 muestra un esquema simple en GeoMondrian. El mismo incluye un cubo llamado *Cáncer*, que a su vez contiene la jerarquía *Distrito* → *Cantón* → *Provincia* como parte de su dimensión *Distritos*. Además se presentan dos medidas dentro del cubo: *Casos* y *Muertes*.

Los parámetros indicados dentro de la etiqueta que comienza con *Level* en la definición del nivel son los encargados de señalar su nombre, su ubicación (columna de la tabla) y tipo. Por ejemplo, en la Figura 6, el componente descriptivo del nivel *Provincia* (fila 7), es de tipo *String* y está en la columna *nombre_provincia* de la tabla *cancer_fact* (fila 3). Un nivel también puede contener un componente espacial, que se agrega por medio de una propiedad adicional. Para ello, se utiliza la etiqueta *Property* dentro del nivel, junto con un nombre, la columna en la tabla y su tipo. Por ejemplo, en la Figura 6, el componente espacial del nivel *Provincia* se llama *geom*, se encuentra en la columna *geometría_provincia* y es de tipo *Geometry* (fila 8).

```
1  <Schema>
2    <Cube name="Cancer">
3      <Table name="cancer_fact"/>
4        <Dimension name="Distritos"
         foreignKey="codigo_distrito">
5          <Hierarchy hasAll="false" primaryKey="
           codigo_distrito">
6            <Table name="geografia"/>
7            <Level name="Provincia"
             column="nombre_provincia" type="String" />
8              <Property name="geom"
               column="geometria_canton" type="Geometry"
               />
9            <Level name="Cantón"
             column="nombre_canton" type="String" />
10             <Property name="geom"
               column="geometria_canton" type="Geometry"
               />
11           <Level name="Distrito"
             column="nombre_distrito" type="String" />
12             <Property name="geom"
               column="geometria_canton" type="Geometry"
               />
13         </Hierarchy>
14       </Dimension>
15     <Measure name="Casos" column="casos"
       aggregator="sum"/>
16     <Measure name="Muertes" column="muertes"
       aggregator="sum"/>
17   </Cube>
18 </Schema>
```

Fig. 6. Esquema simple de GeoMondrian.

Durante el proceso de implementación, es posible que se encuentren problemas para representar adecuadamente ciertas situaciones relacionadas con las necesidades del proyecto. Como alternativa de solución ante ello, la herramienta permite definir estructuras complejas, como dimensiones compartidas, cubos virtuales y medidas calculadas.

C. *Dimensiones compartidas*

La necesidad por compartir dimensiones entre dos o más cubos resalta en sistemas complejos de almacenes de datos y sistemas OLAP espaciales. Por ejemplo, el caso práctico descrito en el presente artículo contiene dimensiones y jerarquías sobre el territorio que son utilizadas por todos los cubos.

La Figura 6 muestra una dimensión *Distritos*, definida dentro del cubo *Cáncer*. Si se incluyera otro cubo que también utilizara *Distritos*, la dimensión

tendría que ser nuevamente definida dentro del cubo recién agregado para que este la conozca. Esto significa que la especificación de una dimensión tendría que realizarse tantas veces como la cantidad de cubos en que se utiliza. Para evitar esa repetición de definiciones en esquemas avanzados, GeoMondrian permite definir dimensiones fuera del cubo facilitando sus reutilizaciones y a la vez simplificando la definición del esquema.

```
1  <Schema>
2    <Dimension name="Distritos"
     foreignKey="codigo_distrito">
3      <!--Misma definición de dimensión Distritos de Figura
       6.-->
4    </Dimension>
5    <Cube name="Cancer">
6      <!--…-->
7      <DimensionUsage name="Distritos" source="Distritos"
       foreignKey="distrito"/>
8      <!--…-->
9    </Cube>
10   <Cube name="Defunciones">
11     <!--…-->
12     <DimensionUsage name="Distritos" source="Distritos"
       foreignKey="distrito"/>
13     <!--…-->
14   </Cube>
15 </Schema>
```

Fig. 7. Uso de dimensiones compartidas en un esquema de GeoMondrian.

La Figura 7 muestra un ejemplo del uso de dimensiones compartidas, donde la dimensión *Distrito* se define una sola vez afuera de la definición de los cubos (filas 2-4) y es compartida entre los cubos *Cáncer* y *Defunciones*. Estos cubos la referencian utilizando una sentencia con la etiqueta *DimensionUsage,* a través de la relación entre la llave foránea *distrito* (de la tabla de hechos) con la llave primaria *codigo_distrito*, especificada en la definición de la dimensión (filas 7 y 12).

D. Cubos virtuales

Un cubo virtual es aquel compuesto por un subconjunto de medidas y dimensiones de uno o más cubos base. Su característica principal es la capacidad para crear un ambiente compartido, propiciando así la mezcla de medidas y dimensiones que antes eran totalmente independientes unas de otras, dada la separación entre los cubos. Por ejemplo, es posible incluir la medida *Tasa de incidencia* dentro de un cubo virtual que utilice las medidas *Cantidad de habitantes* y *Casos*, tomadas de otros cubos físicos, y poder calcular correctamente su valor. Una ventaja adicional que brinda GeoMondrian es que almacena únicamente la definición del cubo en el esquema y no los datos involucrados en el mismo. Por lo tanto, es posible crear diferentes combinaciones y variantes de cubos existentes sin agotar espacio adicional [19].

Para crear el cubo virtual dentro de un esquema, se deben especificar los cubos que se van a combinar, las dimensiones que se van a utilizar y, en este caso particular, las fórmulas para las medidas calculadas. El esquema de la Figura 8 muestra un cubo virtual llamado *cancer_poblacion*, que contiene dimensiones (*Tipo Cáncer* y *Distritos*) y medidas (*Casos* y *Muertes*) del cubo *Cáncer*, así como la medida *Cantidad de habitantes* del cubo *Población.* Su objetivo es posibilitar la creación de la medida *Tasa de incidencia*, generada a partir de medidas que son parte de cubos distintos.

E. Tipos de medidas

1) Medidas calculadas

Las medidas calculadas son aquellas que no están físicamente en el almacén de datos, pero que pueden ser calculadas a partir de otras existentes; ergo, sus valores no están en una columna de la tabla de hechos, sino que son el resultado de una fórmula aplicada. En el esquema, una medida calculada se define dentro del cubo que contiene las medidas que utiliza en su fórmula; sin embargo, en caso de que el cálculo involucre medidas de distintos cubos, es necesario utilizar un cubo virtual.

```
1  <VirtualCube name="cancer_poblacion">
2    <CubeUsages>
3      <CubeUsage cubeName="cancer"
       ignoreUnrelatedDimensions="true"/>
4      <CubeUsage cubeName="poblacion"/>
5    </CubeUsages>
6    <VirtualCubeDimension cubeName="cancer" name="Tipo
     Cáncer"/>
7    <VirtualCubeDimension cubeName="cancer"
     name="Distritos"/>
8    <!--Otras dimensiones necesarias. -->
9    <VirtualCubeMeasure cubeName="poblacion"
     name="[Measures].[Cantidad de habitantes]"/>
10   <VirtualCubeMeasure cubeName="cancer"
     name="[Measures].[Casos]"/>
11   <VirtualCubeMeasure cubeName="cancer"
     name="[Measures].[Muertes]"/>
12   <CalculatedMember name="Tasa de incidencia "
```

```
    dimension="Measures">
13      <Formula>
          <!--Verificación de que el valor para "Cantidad de
          habitantes" sea mayor a cero y diferente de nulo-->
          [Measures].[Casos]/(Aggregate({[Tipo Cáncer].[All
          Tipos]}, [Measures].[Cantidad de
          habitantes])*100000
        </Formula>
14  </CalculatedMember>
15  <!--…-->
16 </VirtualCube>
```

Fig. 8. Uso de cubos virtuales en un esquema de GeoMondrian.

La Figura 8 muestra la definición de la medida calculada *Tasa de incidencia* (filas 12-14) perteneciente al cubo virtual *cancer_poblacion*. Su especificación en el esquema se hace por medio de la cláusula *CalculatedMember*, donde el parámetro *name* se utiliza para especificar el nombre, mientras que el parámetro *dimension* con el valor *Measures* indica que la nueva medida forma parte del grupo de medidas disponibles en el cubo. Asimismo, la cláusula *Formula* es necesaria para definir el cálculo de la medida, que se define en lenguaje MDX. En el caso de *Tasa de incidencia*, el mismo se realiza al dividir los *Casos* entre *Cantidad de habitantes* y multiplicarlos por 100000, con lo cual se obtiene la cantidad de casos de cáncer por cada cien mil habitantes.

Al momento de realizar una consulta que incluye una medida calculada, GeoMondrian obtiene el valor de cada una de las medidas dentro de la fórmula y les aplica el filtro que se haya especificado. Este comportamiento puede generar problemas cuando se trabaja con medidas pertenecientes a cubos con distinta granularidad. Por ejemplo, si se define una medida *Tasa* como *Casos* entre *Cantidad de habitantes* y se realiza una consulta de su valor para *Leucemia* en el *distrito A*, la herramienta trataría de obtener (1) el valor de *Casos* de *Leucemia* en el *distrito A* y (2) el valor de *Cantidad de habitantes* de *Leucemia* en el *distrito A*. Sin embargo, al no poder agregar *Cantidad de habitantes* sobre la dimensión *Tipo Cáncer*, el segundo cálculo fallaría y por lo tanto, también lo haría el de *Tasa*. El problema se da porque el cubo del que se origina *Casos* (*Cáncer*) tiene un mayor nivel de granularidad que el que contiene *Cantidad de habitantes* (*Población*). Como puede ser constatado en la Figura 2, el cubo *Cáncer* cuenta con una dimensión más que Población: *Tipo Cáncer*; eso indica que los datos de *Cáncer* pueden ser filtrados por *Tipo Cáncer*, mientras que los del cubo *Población* no.

Teniendo en cuenta estas condiciones, se especificó ignorar cualquier filtro hecho sobre la dimensión *Tipo Cáncer* al calcular el valor de la medida *Cantidad de habitantes* dentro de la fórmula de *Tasa de incidencia*. En términos prácticos, se aplicó la función MDX *Aggregate* sobre el elemento *All Tipos* de *Tipo Cáncer* para la medida *Cantidad de habitantes*, como se aprecia en la Figura 8 (fila 13). *All Tipos* es el miembro superior de la jerarquía asociada a *Tipo Cáncer*, por lo que su valor para la medida *Cantidad de habitantes* es el resultado de la agregación de todos sus elementos.

Por lo tanto, sin importar los filtros especificados en las consultas, *Cantidad de habitantes* se calcula como su valor total para todos los tipos. Esto hace que, en cualquier caso donde se involucre a la medida *Tasa de incidencia*, GeoMondrian calcule el valor de *Cantidad de habitantes* considerando únicamente los filtros en las dimensiones para las cuales *Cantidad de habitantes* puede agregarse. Por ejemplo, si se pide *Tasa de incidencia* para *Leucemia* en el *distrito A*, la herramienta obtiene (1) el valor de *Cantidad de habitantes* en el *distrito A* y (2) el valor de *Casos* de *Leucemia* en el *distrito A*. Una vez que se tienen esos cálculos, es posible obtener el resultado para la *Tasa de incidencia*. Esta solución implementada en el proyecto permite comprobar que en GeoMondrian es posible construir medidas calculadas a partir de medidas con distinta granularidad.

2) Uso de funciones de agregación

La agregación se relaciona con el cálculo llevado a cabo con respecto a medidas al momento de realizar operaciones como *roll-up* sobre los niveles de una jerarquía. Al agregar medidas como *Casos* sobre las jerarquías de territorio, se utilizan los valores agregados del nivel inferior para realizar la suma y obtener el agregado del nivel superior. Por ejemplo, se suman los casos de los distritos para obtener los casos de los cantones y, una vez calculados, se suman los casos de los cantones para obtener el resultado en las provincias.

Cuando se define una medida, GeoMondrian requiere que se especifique una función de agregación; por ejemplo, en el esquema de la Figura 6 (fila 16) se especifica *SUM* para indicar que los datos de la medida *Muertes* del cubo *Cáncer* se deben sumar al ser agregados (fila 16). No obstante, medidas como *Tasa de incidencia* producen resultados semánticamente incorrectos si son procesadas de esa misma manera, debido a que sus valores agregados no pueden calcularse a partir la

suma simple de los valores de sus niveles inferiores. Con base en estos criterios, surge la siguiente clasificación [5]:

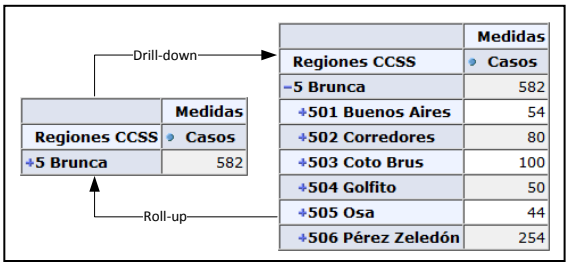
- Medida aditiva: es posible sumarla sobre todas las dimensiones. Las medidas *Casos* y *Muertes* del cubo *Cáncer* son aditivas, utilizan la suma para agregar los valores en cada una de sus dimensiones.
- Medida semiaditiva: aquella que tiene sentido lógico sumar sobre algunas, pero no todas, las dimensiones. La medida *Cantidad de habitantes* del cubo *Población* es semiaditiva, ya que tiene sentido lógico usar la suma para agregarla en todas las dimensiones excepto *Tiempo*. Por ejemplo, si se tuviera un nivel *Quinquenio* en *Tiempo* y se consultara el valor de *Cantidad de habitantes* para cada elemento de él usando la suma como función de agregación, se obtendrían valores semánticamente incorrectos; el quinquenio *2000-2004* devolvería el valor de los cinco años sumados, cuando lo adecuado sería que mostrara el valor correspondiente al año más reciente (*2004*).
- Medida no aditiva: no es posible sumarla sobre ninguna de las dimensiones. *Tasa de incidencia* es un ejemplo de una medida no aditiva, debido a que la aplicación de suma al agregar los valores en cualquier dimensión genera un resultado semánticamente incorrecto.

Actualmente, la herramienta no incluye mecanismos destinados específicamente al manejo de medidas no aditivas y semiaditivas. Sin embargo, existen funciones de agregación que pueden ser utilizadas para representar las medidas no aditivas, como *MAX*, *MIN* y *AVG*. Otra forma de utilizarlas es a través de medidas calculadas, siempre que estas puedan ser adaptadas al caso particular.

F. *Operaciones de análisis y despliegue de resultados*

En cuanto a operaciones OLAP, GeoMondrian permite las operaciones básicas de:

- R*oll-up* y *drill-down*, como se presenta en la Figura 9. Cuando se aplica *drill-down* sobre el elemento *5 Brunca*, se muestran los miembros del nivel inferior. Por el contrario, esos sub-elementos de *Brunca* se ocultan cuando se les aplica *roll-up*.



	Medidas
Regiones CCSS	Casos
+5 Brunca	582

	Medidas
Regiones CCSS	Casos
-5 Brunca	582
+501 Buenos Aires	54
+502 Corredores	80
+503 Coto Brus	100
+504 Golfito	50
+505 Osa	44
+506 Pérez Zeledón	254

Fig. 9. Drill-down y roll-up en GeoMondrian

- *Pivot*, como se aprecia en la Figura 10. Cuando se hace *pivot*, los ejes cambian. Por lo tanto, las medidas se mueven hacia la izquierda de la tabla y los elementos de *Regiones CCSS* se trasladan a la parte superior.

	Medidas	
Regiones CCSS	Casos	Muertes
+1 Central Norte	2.339	1.190
+2 Central Sur	3.013	1.415
+3 Chorotega	459	245

Pivot

	Regiones CCSS		
Medidas	+1 Central Norte	+2 Central Sur	+3 Chorotega
Casos	2.339	3.013	459
Muertes	1.190	1.415	245

Fig. 10. *Pivot* en GeoMondrian

- *Slice-and-dice*, que se muestra en la Figura 11. Se seleccionan únicamente el año *2005* y dos regiones: *3 Chorotega* y *4 Pacífico Central*. También muestra cómo es posible combinar dimensiones (*Regiones INEC* y *Tiempo*) y medidas (*Casos* y *Muertes*) en una misma consulta.

		Medidas	
Regiones INEC	Tiempo	Casos	Muertes
+3 Chorotega	2005	459	245
+4 Pacífico Central	2005	359	192

Fig. 11. *Slice-and-dice* en GeoMondrian

VII. DESPLIEGUE MEDIANTE HERRAMIENTAS GRÁFICAS

En el ambiente de *software* libre, la variedad de herramientas disponibles de tipo SOLAP para el

análisis y trazado de datos espaciales en mapas es escasa. Esa limitación generó la búsqueda de otras opciones que proporcionaran las funciones necesarias. Finalmente se seleccionó GeoOLAP[1], una herramienta desarrollada originalmente por Camptocamp[2]. A pesar de que el proyecto ya no continúa siendo impulsado por la organización, ofrece algunas de las operaciones básicas que, en conjunto con visualización de mapas, permiten representar los datos gráficamente durante el análisis. Para su funcionamiento, GeoOLAP requiere la definición de los cubos, tal como se indicó en la sección VI.

A. Despliegue de dimensiones y medidas

La principal característica de GeoOLAP es su capacidad para manejar y representar dimensiones espaciales, lo cual realiza a través de la extracción de datos correspondientes a coordenadas que luego representa sobre un mapa. La herramienta también permite combinar dimensiones espaciales y convencionales en una misma consulta, filtrando los datos según se le indique. Además, despliega los datos de tres maneras distintas al mismo tiempo: gráficos, mapas y tablas; como puede ser visto en la Figura 12, que muestra el despliegue de los resultados generados tras una consulta de *Casos* por *Sexo* en las *Provincias* en el cubo *Cáncer*.

Sin embargo, GeoOLAP también tiene sus limitaciones. Entre las más evidentes, el sistema sólo ejecuta las consultas que contengan al menos una dimensión espacial y no incluye opciones para seleccionar más de una medida en una consulta. Si bien estos factores la ponen en desventaja en relación con herramientas como GeoMondrian, su uso se justifica en la posibilidad de visualizar los datos sobre mapas.

Un ejemplo del despliegue de los elementos espaciales sobre el mapa puede ser apreciado en la parte superior de la Figura 12. La escala de colores utilizada se crea dinámicamente con cada consulta, tomando como base los valores de la medida consultada. Esto significa que se crean clases (equivalentes a un rango de valores) a las cuales se les asigna un color determinado, de manera que la escala de los mismos corresponda con la incidencia de la medida seleccionada. Los elementos pertenecientes a las clases generadas son llamados coropletas y su simbología aparece a la derecha del mapa. En la Figura 13 se presenta el detalle de las clases de las coropletas, donde cada una es representada por un color distinto.

B. Operaciones

Durante la investigación se encontraron herramientas que ofrecen capacidades de navegación OLAP completas, pero que carecen de visualización en mapas; mientras que otras tienen la opción de mostrar resultados en mapas, pero no ofrecen todas las operaciones y funcionalidades OLAP requeridas para un análisis robusto. GeoOLAP significó la opción con mejor balance entre ambos requerimientos, dado que soporta algunas de las operaciones OLAP básicas y maneja datos espaciales. Entre sus principales operaciones están:

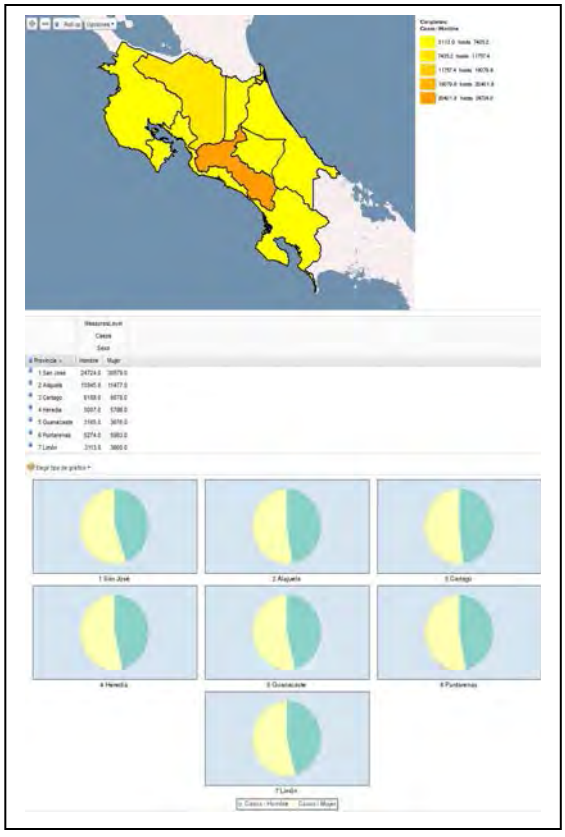


Fig. 12. *Despliegue de resultados en GeoOLAP*

[1] P. Mauduit (2012, Nov.). *"pmauduit/GeoBI – GitHub"*, https://github.com/pmauduit [Online]. Available: https://github.com/pmauduit/GeoBI [Apr. 30, 2013].

[2] Y. Jacolin and A. Gioia (2010, Nov. 9). *"GeoBI Initiative: The open source location intelligence ecosystem"*, http://www.spagoworld.org [Online]. Available: http://www.spagoworld.org/spw-resources/Resources/Presentations/GeoBI@fOSSa2010.pdf [Apr. 30, 2013].

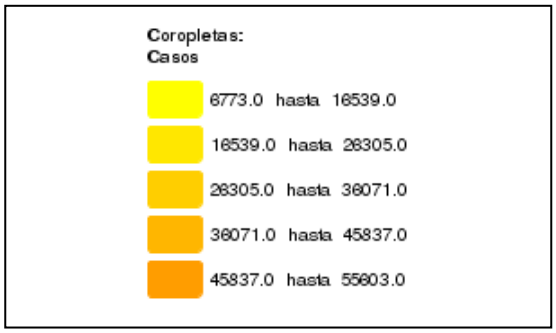


Fig. 13. Coropletas en GeoOLAP.

- *Drill-down* y *roll-up*. *Drill-down* se realiza al pulsar sobre una región geográfica, mientras que *roll-up* se ejecuta al oprimir el botón correspondiente en la parte superior del mapa. Por ejemplo, si se están desplegando los cantones de *San José* sobre el mapa, la acción de *roll-up* mostrará sólo esa provincia (sube un nivel), como se observa en la Figura 14. *Drill-down* genera el efecto contrario.

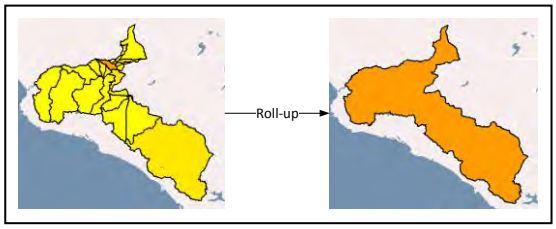


Fig. 14. Roll-up en GeoOLAP.

- *Slice-and-dice*. La figura 15 presenta el resultado que se genera al seleccionar sólo los datos correspondientes a *3 Cartago* y *4 Heredia*.

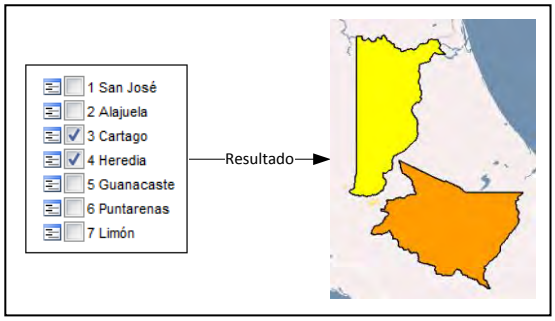


Fig. 15. Selección de miembros específicos de una dimensión en GeoOLAP (slice-and-dice).

- Combinación de dimensiones espaciales. No es una operación OLAP típica, pero se trata de una opción adicional que ofrece GeoOLAP. En determinado caso, se combinan las geometrías correspondientes a ambas dimensiones y se despliega un híbrido de las mismas sobre el mapa. Este proceso implica la creación de nuevos elementos a partir de la combinación de las dimensiones espaciales. Por ejemplo, la Figura 16 muestra el mapa producido al seleccionar la medida *Casos* del cubo *Cáncer* para el miembro *6 Puntarenas* del nivel *Provincias* y dos miembros de *Regiones INEC*: *4 Pacífico Central y 5 Brunca*. La ejecución de la consulta crea dos nuevos elementos para propósitos de despliegue: (1) *6 Puntarenas 4 Pacífico Central* (color naranja en el mapa) y (2) *6 Puntarenas 5 Brunca* (color amarillo en el mapa).

C. Estilos para la visualización de resultados

Existe la posibilidad de cambiar varias opciones relacionadas con el despliegue de las coropletas y otros símbolos adicionales. Para ello, se presentan dos paneles específicos en GeoOLAP: el llamado *Coropletas* permite seleccionar el método de clasificación de las mismas (intervalos iguales, valores únicos o cuantilos), su rango de colores y la cantidad de clases; mientras que el otro, llamado *Símbolos adicionales*, permite desplegar símbolos proporcionales sobre el mapa.

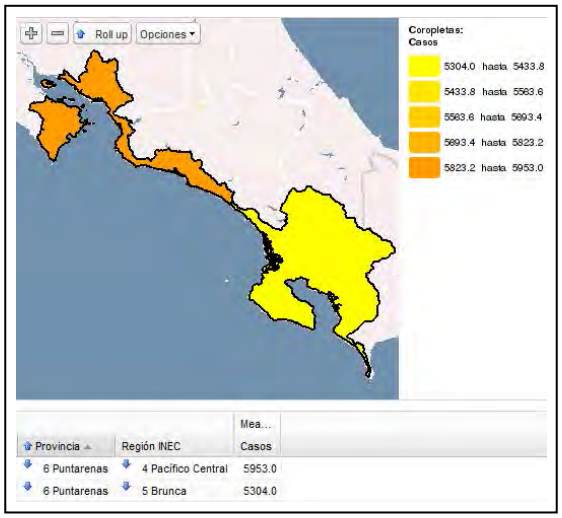


Fig. 16. Mapa que combina dimensiones espaciales en GeoOLAP.

La interfaz permite elegir el tamaño del símbolo adicional y su indicador asociado. Además, presenta la opción de poder mostrar alguno de los dos tipos de símbolos: gráficos de barras y gráficos circulares. Un ejemplo de despliegue de los casos de cáncer por sexo en cada provincia se presenta en la Figura 17. Los símbolos adicionales están activados, por lo que sobre cada provincia se muestra el gráfico circular correspondiente a la distribución de casos de cáncer por sexo en esa área geográfica. En este ejemplo, se despliegan tanto las coropletas como los símbolos. La simbología puede observarse a la derecha, debajo de las coropletas.

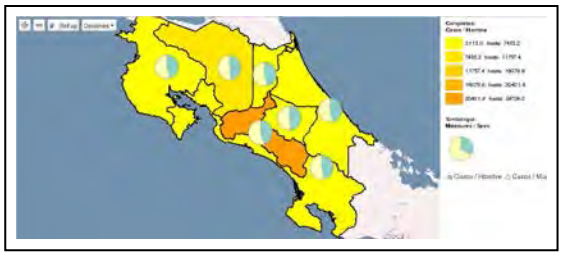


Fig. 17. Mapa con gráficos circulares y coropletas en GeoOLAP.

A pesar de que sus funciones están centradas en el uso del mapa, GeoOLAP también ofrece el despliegue de los datos a través de tablas y gráficos. Ambas se muestran como secciones separadas bajo el área del mapa, lo cual puede ser visto en la Figura 12. Las operaciones de análisis ejecutadas se traducen en cambios automáticos en estas dos representaciones, que están siempre presentes.

La tabla que se puede ver en la Figura 12, muestra los datos en su versión más convencional, algo similar al estilo de la que utiliza GeoMondrian, pero más limitada en cuanto a opciones. Por ejemplo, no cuenta con la operación *pivot*. Sin embargo, también puede afectar el mapa porque cuenta con funciones que hacen posible ejecutar *drill-down* y *roll-up* sobre sus miembros. Asimismo, el sistema permite seleccionar un tipo de gráfico de cuatro existentes: circular por columna, circular por fila, barras horizontales y barras verticales. Un gráfico circular por fila se muestra en la parte inferior de la Figura 12. Es importante señalar que los símbolos desplegados sobre el mapa no tienen relación directa con los gráficos que se presentan debajo de él. Son completamente independientes, lo cual significa que pueden estarse desplegando gráficos de barras en la sección exclusiva de gráficos y al mismo tiempo gráficos circulares como símbolos adicionales sobre el mapa.

VIII. OTROS DESAFÍOS EN EL USO Y ANÁLISIS DE DATOS ESPACIALES

Los desafíos relacionados con el uso de datos espaciales se extienden más allá del diseño de cubos SOLAP. Por lo tanto, es posible que surjan nuevos retos durante el proceso de implementación e integración con herramientas cliente. En el caso práctico descrito, el uso de *shapefiles* para la representación de geometrías trajo ciertos desafíos al proyecto. Por ejemplo, dado que la geometría de un polígono se define como un conjunto de puntos, el espaciado entre cada par de ellos determina la escala de detalle en la representación de la geometría; consecuentemente, un mal uso de esta escala puede causar que la representación se torne irregular o inconsistente. Asimismo, entre más nivel de detalle se desee, más puntos se necesitan en la representación, lo cual a su vez requiere mayor espacio de almacenamiento y mayor tiempo de trazado.

La definición correcta de las geometrías es de suma importancia para lograr las intersecciones necesarias entre las geometrías y, por ende, el despliegue correcto de la información. A pesar de que existen sistemas automatizados que ayudan a simplificar y limpiar la definición de las geometrías, el proceso puede ser arduo cuando se tienen múltiples dimensiones espaciales.

Cuando se inició la implementación del proyecto, se contaba con *shapefiles* que contenían un alto nivel de detalle para las geometrías. Esto provocaba que geometrías de regiones muy irregulares tomaran mayor tiempo de despliegue. De igual forma, al momento de combinar dimensiones espaciales, la intersección de geometrías ocasionaba problemas durante el despliegue de los resultados. Por lo tanto, fue necesario simplificar las geometrías para eliminar estos problemas.

En cuanto a la herramienta cliente, debido al requerimiento de uso de *software* libre, el proceso de búsqueda fue extenso. La necesidad de una herramienta que ofreciera operaciones OLAP y el despliegue de resultados en mapas, gráficos y tablas en forma sincronizada, resultó retador. Tal y como se mencionó en la sección II, la mayor parte de las herramientas SOLAP son propietarias, mientras que las opciones *open-source* disponibles cuentan con escasa documentación y suelen presentar limitaciones en su funcionalidad. Adicionalmente, existen pocos recursos de apoyo al momento de trabajar con *software* libre SOLAP.

IX. CONCLUSIONES

Se ha comprobado que gran parte de los datos almacenados en bases de datos poseen una característica que hace posible asociarlos con un área geográfica específica, la cual se presenta comúnmente a través de un nombre; por ejemplo, el de un estado, una región o un país [5]. Sin embargo, la representación descriptiva del dato no es suficiente para efectuar un análisis espacial, porque también se requiere la capacidad de manejar componentes espaciales. La necesidad de encontrar una solución que integrara ambas características, sumada al crecimiento en la cantidad de datos espaciales disponibles, provocó el surgimiento de nuevas tecnologías como SOLAP. La utilización de herramientas para llevar a cabo el análisis de los datos espaciales ha tomado preponderancia tras los recientes avances en áreas como Sistemas de Posicionamiento Global (GPS – *Global Positioning System*) y Sistemas Globales de Navegación por Satélite (GNSS – *Global Navigation Satelite System*).

Actualmente existen diversas soluciones de *software* propietario que ofrecen alternativas SOLAP. No obstante, en el ambiente *open-source* las opciones disponibles son más limitadas. GeoMondrian maneja datos espaciales, pero no contiene opciones de visualización, mientras que GeoOLAP permite la visualización de datos espaciales sobre mapas, pero carece de ciertas operaciones básicas OLAP, como *pivot*.

La representación de ciertas situaciones relacionadas con los requerimientos del proyecto podría resultar difícil de plasmar en un sistema SOLAP. Sin embargo, las herramientas encontradas incluyen estructuras avanzadas como alternativas para resolver ese tipo de problemas. Entre ellas están las dimensiones compartidas, las medidas calculadas, las distintas funciones de agregación y los cubos virtuales.

Los cubos virtuales hacen que medidas complejas (como *Tasa de incidencia*) sean calculables y generan beneficios de espacio; empero, ha sido anunciado que Mondrian 4 (aún en desarrollo) prescindirá de ellos. Esta nueva versión de la herramienta que dio origen a GeoMondrian usará estructuras distintas para generar la misma funcionalidad. Por ejemplo, ofrecerá la posibilidad de manejar más de una tabla de hechos en el mismo cubo a través de la utilización de *MeasureGroups*.

A pesar de las limitaciones encontradas, se comprobó que es posible afrontar variadas necesidades de análisis a través de diferentes herramientas disponibles. Mediante la utilización de las mismas, se les brindan nuevas opciones a los usuarios no expertos, quienes pueden adaptar la solución espacial de BI *open-source* a sus requerimientos y considerarla así como parte indispensable en la toma de decisiones. Tomando en cuenta la posible extensión de las funcionalidades a partir del *software* existente, el campo aún permanece abierto para el desarrollo de una herramienta SOLAP completa y gratuita.

REFERENCIAS

[1] PostGIS. *"PostGIS - Spatial and Geographic Objects for PostgreSQL"*, http://postgis.net [Online]. Available: http://postgis.net [Mar. 26, 2013].

[2] Asamblea Legislativa de la República de Costa Rica. *"Páginas – Detalle Proyectos de Ley"*, http://www.asamblea.go.cr [Online]. Available: http://www.asamblea.go.cr/Centro_de_Informacion/Consultas_SIL/Pginas/Detalle%20Proyectos%20de%20Ley.aspx?Numero_Proyecto=16912 [Jun. 9, 2012].

[3] Microsoft Developer Network. "*Consulta de MDX básica (MDX)"*, http://msdn.microsoft.com/es-es [Online]. Available: http://msdn.microsoft.com/es-es/library/ms144785.aspx [Mar. 26, 2013].

[4] C. Traynor and M. G. Williams, "Why Are Geographic Information Systems Hard to Use?" in *CHI '95: Conference Companion on Human Factors in Computing Systems*, Denver, CO, 1995, pp. 288-289.

[5] E. Malinowski and E. Zimányi, *Advanced Data Warehouse Design: From Conventional to Spatial and Temporal Applications*, New York: Springer, 2008, pp. 4-179.

[6] Environmental Systems Research Institute (1998, Jul.), ESRI Shapefile Technical Description: An ESRI White Paper – July 1998. [Online]. Available: http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf [Mar. 26, 2013].

[7] Environmental Systems Research Institute (1998, Mar.), Spatial Data Warehousing: An ESRI White Paper – March 2008 [Online]. Available: http://spatialnews.geocomm.com/whitepapers/datawarehouse1.pdf [Mar. 26, 2013].

[8] D. Lean. (2008, Nov. 1) "SQL 2008 Spatial Samples, Part 2 of 9 - Background on Spatial Types & Well Known Text (WKT)" [Weblog entry]. *Dave does Data.* Microsoft Developer Network. Available: http://blogs.msdn.com/b/davidlean/archive/2008/11/01/sql-2008-spatial-samples-part-2-of-n-background-on-spatial-types-well-known-text-wkt.aspx [Mar. 26, 2013].

[9] OpenGeo. "*Introduction to PostGIS - Section 8: Geometries",* http://workshops.opengeo.org [Online]. Available: http://workshops.opengeo.org/postgis-intro/geometries.html [Mar. 26, 2013].

[10] J. da Silva, A. C. Salgado, and V. C. Times, "An open source and web based framework for geographic and multidimensional processing," in *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, Dijon, France, 2010, pp. 63-67.

[11] JPivot. *"JPivot – Home",* http://jpivot.sourceforge.net [Online]. Available: http://jpivot.sourceforge.net [Apr. 30, 2013].

[12] Pentaho Mondrian Project . *"Pentaho Mondrian Documentation - MDX Specification",* http://mondrian.pentaho.com [Online]. Available: http://mondrian.pentaho.com/documentation/mdx.php [Mar. 26, 2013].

[13] M. Scotch and B. Parmanto, "SOVAT: Spatial OLAP Visualization and Analysis Tool" in *HICSS '05: Proceedings of the 38th Hawaii International Conference on System Sciences*, Big Island, HI, 2005.

[14] PostGIS. *"PostGIS 2.0 Manual",* http://postgis.net [Online]. Available: http://postgis.net/docs/manual-2.0/index.html [Apr. 11, 2013].

[15] SpagoWorld. *"SpagoBI - BI components"*, http://www.spagoworld.org [Online]. Available: http://www.spagoworld.org/xwiki/bin/view/SpagoBI/BIComponents [Apr. 3, 2013].

[16] S. Bimonte, A. Tchounikine, and M. Miquel, "Spatial OLAP: Open Issues and a Web Based Prototype" in *10th AGILE International Conference on Geographic Information Science*, Aalborg, Denmark, 2007.

[17] S. Rivest, Y. Bédard, M.-J. Proulx, M. Nadeau, F. Hubert, and J. Pastor. (2005, Dec.). "SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data", *ISPRS Journal of International Society for Photogrammetry and Remote Sensing*, vol. 1 (60), pp. 17-33, 2005.

[18] Stratebi Business Solutions. *"STPivot",* http://www.stratebi.com [Online]. Available: http://www.stratebi.com/stpivot. [Apr. 12, 2013].

[19] Microsoft Developer Network. *"Virtual Cubes"*, http://msdn.microsoft.com/es-es [Online]. Available: http://msdn.microsoft.com/en-us/library/aa216377%28v=sql.80%29.aspx [Mar. 26, 2013].

[20] Y. Bédard, T. Merret, and J. Han, "Fundamentals of spatial data warehousing for geographic knowledge discovery" in *Geographic Data Mining and Knowledge Discovery*, H. J. Miller and J. Han, Eds. New York: Taylor & Francis, pp. 53-73, 2001.

[21] Y. Bédard, E. Bernier, S. Larrivée, M. Nadeau, M.-J. Proulx y S. Rivest (2009, Nov.). *"Spatial OLAP",* http://www.spatialbi.com [Online]. Available: http://www.spatialbi.com [Apr. 2, 2013].