

# Un Modelo Unificado de Visualización

Sergio Martig Silvia Castro Pablo Fillottrani Elsa Estévez  
Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur  
Avda. Alem 1253 – 8000 Bahía Blanca  
{srm,smc,prf,ece}@cs.uns.edu.ar

## Resumen

En este trabajo presentamos un Modelo Unificado de Visualización que tiene en cuenta las características propias de todas las áreas de la Visualización. Nuestra propuesta consiste en un único modelo en el cual se pueda enfocar tanto en los procesos como en los estados de los datos y que es aplicable a cualquier visualización independientemente del campo particular de origen. En este modelo quedan explícitamente representadas las operaciones provistas, los operandos sobre los que se pueden aplicar, conjuntamente con la secuencia de transformaciones propias del proceso. Un modelo tal constituye un marco conceptual en el que se pueden definir las interacciones necesarias, pudiéndose determinar sobre *quién* se opera, *qué* resultados se obtienen y *cómo* impacta sobre el proceso general.

Por otro lado este modelo es el punto de partida para la creación de una arquitectura de referencia que sirva tanto para que los investigadores dispongan de un conjunto de conceptos comunes sobre los cuales obtener un mejor entendimiento de los problemas conceptuales y técnicos del área, como para que los desarrolladores puedan implementar ambientes que compartan una plataforma común.

**Palabras Clave:** Sistemas de Visualización, Visualización, Arquitecturas de Software. WCGV.

## 1 Introducción

Mientras las nuevas tecnologías tanto en hardware como en software contribuyen a un aumento en el volumen y la diversidad de los datos que las organizaciones manejan, la exploración y la visualización de estos datos se tornan cada vez más dificultosas. Por esto, y debido al crecimiento en la cantidad y variedad de usuarios y a la demanda de mayor funcionalidad, es deseable que los sistemas de Visualización de Información (VI) cumplan con propiedades como la extensibilidad, la modificabilidad, la predictibilidad y la posibilidad de estar integrados por componentes intercambiables.

En los últimos años han sido presentados numerosos ambientes de VI; además, existen paquetes estándares que implementan la mayoría de las técnicas de visualización conocidas. A pesar de que estos sistemas no cumplen completamente con las propiedades mencionadas, sus características similares permiten afirmar que el área está madurando lo suficiente como para desarrollar una arquitectura de referencia que sí facilite su cumplimiento. Tal arquitectura de referencia permitiría la especificación e integración de los componentes de hardware y software necesarios para la implementación de sistemas de VI para los diferentes dominios que han surgido recientemente, tales como las consultas a bases de datos, data mining, monitoreo de sistemas, etc.

En ese contexto es necesaria la consolidación de un modelo de referencia para los procesos de visualización. Si bien actualmente existen modelos aplicables a los distintos campos de visualización, no se cuenta con un modelo unificado en todas las áreas de visualización que tenga en cuenta cada una de sus características propias y que pueda tomarse como punto de partida para desarrollar una arquitectura de software. Es por esto que nuestra investigación se enfocó en obtener dicho modelo y éste es el modelo que describimos en este trabajo.

## 2 Características del Proceso de Visualización

Los procesos de visualización pueden ser utilizados en una amplia gama de dominios de aplicación. De la gran variedad de dominios de aplicación de la visualización surgen precisamente los distintos *campos de la disciplina*. Si bien el objetivo común es la obtención de representaciones visuales interactivas con el propósito de ampliar la adquisición y el uso del conocimiento, según la naturaleza y características de la información a visualizar podemos hablar de:

- ▶ *Visualización Científica*: Tiene como característica la visualización de datos científicos, típicamente datos físicos. El sustrato espacial se encuentra presente en los datos a visualizar. Dentro de este campo a su vez se distinguen:
  - Visualización de Volúmenes: Representación, manipulación y rendering de datos volumétricos.
  - Visualización de Flujos: Representación, manipulación y rendering de datos vectoriales y tensoriales.
- ▶ *Visualización de Software*: Incluye la visualización de algoritmos y programas (datos y código). Usualmente se distingue entre:
  - Visualización de Algoritmos: Visualización de las estructuras de alto nivel que caracterizan a un software. Enfoque básicamente pedagógico.
  - Visualización de Programas: Visualización del código y de los datos reales de un programa. Enfoque centrado en herramientas de la ingeniería de software.
- ▶ *Visualización de Información*: Visualización de datos abstractos, no basados en lo físico. Los datos no poseen un mapeo espacial inherente.

En una primera instancia, el diseño de un conjunto de interacciones bien definidas requiere de conocimiento en el dominio específico de la aplicación. Esto es natural ya que cada disciplina y las características particulares de las mismas determinan los análisis a los que se someterán los datos. Afortunadamente, aunque sea frecuente que distintos dominios de aplicación requieran representaciones visuales diferentes, varios de ellos pueden compartir estados intermedios de los datos, o requerir manipulaciones a nivel de vistas similares o incluso necesitar de las mismas transformaciones de datos.

Es claro que los requerimientos impuestos por cada dominio pueden llegar a ser dramáticamente diferentes; sin embargo, como expresáramos previamente, de un análisis cuidadoso de los mismos podemos encontrar denominadores comunes :

- ▶ En los extremos de todos los procesos de visualización se encuentran los datos propios de cada dominio de aplicación, en uno, y una representación abstracta de esos datos en pantalla, en el otro. Las propiedades distintivas de los datos manejados en los extremos del proceso permite hablar de *separación entre datos y vista*.
- ▶ Esa separación entre datos y vista se hace más evidente dentro de la visualización de información dado que los datos a ser mostrados carecen usualmente de un mapeo espacial inherente.
- ▶ Es evidente que entre los dos extremos mencionados los datos van sufriendo sucesivas transformaciones.

- ▶ Por otro lado la visualización tiene que ser el medio por el cual el usuario explora el espacio de datos de su interés, para lograr obtener un *insight* de los mismos. Es claro que un proceso con esas características debe ser netamente interactivo.
- ▶ Para que los usuarios puedan interactuar de manera efectiva debe ser claro y natural sobre qué se está trabajando y cuáles son los efectos de las interacciones posibles propuestas.

Lo expuesto nos conduce, sin lugar a dudas, a la necesidad de contar con un modelo *consistente* para los procesos de visualización que sea válido para los diferentes dominios de aplicación. Esa consistencia permitirá a los usuarios lograr interacciones efectivas en el proceso de visualizar datos provenientes desde distintos dominios de aplicación, basándose en un único modelo mental. De esta manera se facilitará la interacción con el sistema disminuyendo así el *gulf of execution* – “la diferencia entre las intenciones del usuario y las acciones permitidas”.

La existencia de un modelo tal no sólo beneficia a los usuarios, sino también a los diseñadores. Al momento de extender un sistema existente para incluir nuevos dominios de aplicación, los diseñadores poseen un marco de referencia que les define cuáles son las transformaciones que deben sufrir los datos y el conjunto básico de operaciones que deberán proveer.

Lo que se busca básicamente es proveer un modelo que sea consistente con las posibles intenciones de los usuarios y brindar un marco en el que sea claro cómo interactúan las posibles transformaciones y las operaciones provistas. El modelo debe asistir tanto a los usuarios como a los diseñadores en la evaluación y ejecución de las tareas adecuadas para alcanzar las metas propuestas. En otras palabras se debe disminuir el *gulf of evaluation* – la *realimentación provista por el sistema* “sea directamente interpretable en términos de las intenciones y expectativas del usuario”.

Una característica que debe proveer el modelo es permitir que los usuarios puedan enfocar su atención en las *operaciones* necesarias para lograr un determinado resultado, como así también sobre los *operandos* en los diferentes estadios que abarca el proceso. Un ejemplo de modelo enfocado en las operaciones es el pipeline tradicional de visualización en el que el enfoque se realiza fuertemente en el proceso más que en el estado de los datos. En el otro extremo y como ejemplo de un modelo enfocado directamente en los operandos podemos nombrar a las planillas de cálculo, las cuales muestran los distintos estados de los datos manipulados más que las relaciones existentes entre ellos.

Si bien los investigadores en el área han presentado distintos modelos, generalmente éstos estaban enfocados a uno de los campos de visualización en particular y no presentan las características propuestas. Ejemplos de este trabajo previo son el pipeline de W. Schroeder para visualización científica [20], el pipeline presentado por S. Card [5] para visualización de información y el modelo de estados de datos presentado por E. Chi [7]. En ninguno de estos casos se logran conjuntamente los objetivos propuestos.

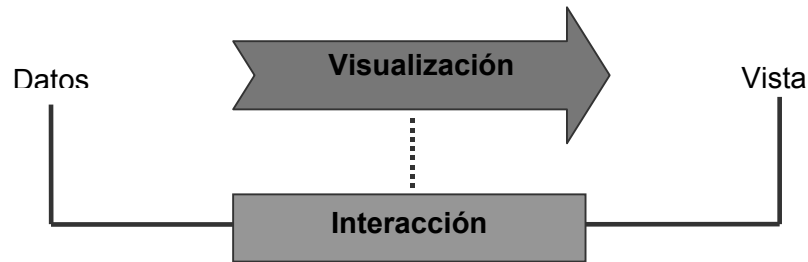
En la próxima sección se presenta el modelo propuesto, describiendo tanto los estados de los datos como los procesos de transformación a lo largo de todo el proceso de visualización. En la sección 4 se dan distintos ejemplos provenientes de diversos campos de visualización en el contexto del modelo presentado. Finalmente se detallan las conclusiones y el trabajo futuro.

### **3 Un Modelo Unificado de Visualización**

Nuestra propuesta consiste en un único modelo en el cual se pueda enfocar tanto en los procesos como en los estados de los datos y que es aplicable a cualquier visualización independientemente del campo particular de origen. En este modelo quedan explícitamente representadas las operaciones provistas, los operandos sobre los que se pueden aplicar, conjuntamente con la secuencia de transformaciones propias del proceso. Un modelo tal constituye un marco conceptual

en el que se pueden definir las interacciones necesarias, pudiéndose determinar sobre *quién* se opera, *qué* resultados se obtienen y *cómo* impacta sobre el proceso general.

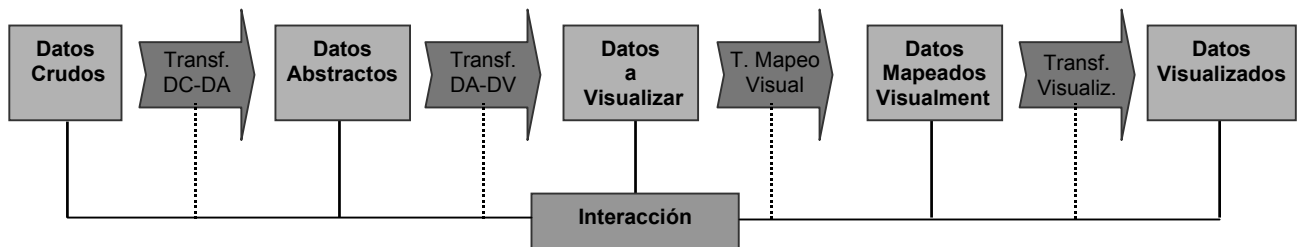
Intuitivamente, cualquier proceso de visualización puede pensarse como una transformación de los datos en una representación visual; es un proceso cognitivo en el cual el usuario tiene que poder interactuar para lograr el objetivo buscado.



**Fig 1: El proceso de visualización**

Por todo lo expuesto vemos que es necesario un modelo en el que podamos describir las transformaciones y modelar los estados intermedios de los datos, para poder enfocarnos sobre los operadores o sobre los operandos favoreciendo así el entendimiento de las interacciones posibles.

Nuestro modelo, es un modelo de estados representado como un flujo entre los distintos estados que van asumiendo los datos a lo largo del proceso. En el esquema planteado, los nodos representan los estados de los datos y las aristas, las transformaciones necesarias para pasar de un estado al próximo. El modelo consiste en cinco estados y cuatro transformaciones que permiten pasar de un estado al próximo (ver Fig. 2).



**Fig. 2: Modelo Unificado de Visualización**

Podemos describir entonces nuestro modelo en términos de los estados y las transformaciones como mostramos en el siguiente cuadro:

Estados	Transformaciones
<p><b>Datos Crudos</b> Son los datos provenientes del dominio de aplicación.</p>	
<p style="text-align: center;"><b>Transf. Datos Crudos – Datos Abstractos</b></p> <p>Esta transformación permite al usuario seleccionar cuáles son los datos que quiere visualizar y llevarlos a un formato manejable por el sistema. Durante esta transformación se generan los metadatos.</p>	

<p><b>Datos Abstractos</b></p> <p>Son los datos potencialmete visualizables por el usuario. Además de los datos propiamente dichos, también se dispone de los metadatos generados en la transformación anterior.</p>	
	<p style="text-align: center;"><b>Transf. Datos Abstractos – Datos a Visualizar</b></p> <p>Esta transformación permite que el usuario indique qué quiere visualizar dentro del conjunto de Datos Abstractos.</p>
<p><b>Datos a Visualizar</b></p> <p>Son todos los datos que van a estar presentes en la visualización. Puede ser un subconjunto de los datos del estado anterior.</p> <p>Es importante notar que para un mismo conjunto de Datos Abstractos se pueden generar varios Datos a Visualizar.</p>	
	<p style="text-align: center;"><b>Transf. de Mapeo Visual</b></p> <p>Esta transformación permite que el usuario especifique cómo quiere visualizar los datos presentes en el conjunto de Datos Visualizar. Se generan las estructuras necesarias para soportar el sustrato espacial, los elementos visuales que se utilizarán en la representación y los atributos de los mismos</p>
<p><b>Datos Mapeados Visualmente</b></p> <p>Son los datos a visualizar enriquecidos con información necesaria para su representación en pantalla.</p> <p>Dado que un mismo conjunto de datos puede ser mostrado de varias maneras distintas en pantalla, resulta natural el hecho de que pueda existir más de un conjunto de Datos Mapeados Visualmente para un conjunto de Datos a Visualizar dado.</p>	
	<p style="text-align: center;"><b>Transf. de Visualización</b></p> <p>Esta transformación permite la presentación en pantalla de todos los datos contenidos en el conjunto de Datos Mapeados Visualmente. Usualmente resultará en la aplicación de una técnica de visualización que soporte el mapeo visual presentado.</p>
<p><b>Datos Visualizados</b></p> <p>Son los productos finales del proceso. Constituyen en realidad el punto de partida para el proceso de exploración que deberá realizar el usuario. Cada representación visual se corresponde con un conjunto de Datos Mapeados Visualmente.</p>	

### 3.1 Los estados de los datos

A lo largo de todo el proceso los datos van sufriendo transformaciones desde su estado en el dominio de aplicación hasta llegar a obtener una representación visual de los mismos.

La distinción entre los estados mostrados no siempre obedece a diferencias estructurales sino también a los diferentes roles que cumplen dentro de la secuenciación del proceso total. En algunos casos, la distinción sólo tiene como justificativo la representación conceptual de un determinado estado, siempre teniendo como objetivo la consistencia con el modelo conceptual del usuario.

Como se puede observar en la Fig. 2 distinguimos los siguientes estados:

- *Datos Crudos*: Se corresponde con el estado inicial. La clasificación de los datos es un tema muy amplio y complejo, por lo que lo abordaremos superficialmente y enfatizando las características que puedan resultar interesantes para su posterior visualización. Bertin (1977) sugiere dos formas básicas de datos: Valores de Datos y Estructuras de Datos. En otras palabras clasifica los datos en *entidades y relaciones*. Estos conceptos tienen una larga historia en el diseño de Bases de Datos pero su uso en Visualización es reciente.

- *Entidades*: Son los objetos de interés que queremos visualizar.

- *Relaciones*: Las relaciones pueden estar definidas por las estructuras y/o patrones bajo los que se presentan las entidades. Las relaciones pueden estar en forma explícita dentro de los Datos Crudos u obtenerlas puede llegar a ser el objetivo del proceso de visualización. Las relaciones pueden ser físicas, conceptuales o estructurales. Tanto las entidades como las relaciones poseen propiedades que se expresan en término de sus atributos.

- *Atributos*: Expresan las propiedades de las entidades y relaciones; no pueden ser pensados en forma independiente. Un aspecto importante a considerar al momento de diseñar una visualización son las características de los atributos a mostrar.

Cuando nos referimos a datos crudos consideramos que éstos pueden provenir no sólo de fuentes externas, sino que también de visualizaciones previas.

- *Datos Abstractos*: Es un estado intermedio de los datos, en el que éstos están en un formato manejable pero aún no visualizable. En este estado tenemos todos los datos que el usuario pre-seleccionó dentro de los Datos Crudos como potencialmente visualizables.

Dentro de este estado podemos tener entonces un subconjunto de los Datos Crudos al cual se le puede haber agregado datos derivados de los existentes o también pueden ser Datos Crudos que se pueden haber mejorado de alguna manera. En síntesis, los Datos Abstractos constituyen el universo visualizable. Cualquier interacción del usuario que exceda a los datos almacenados en este estado demandará una nueva captura de Datos Crudos.

Un aspecto importante a destacar es que en este estado, además de disponer de los datos de interés para el usuario en la forma de datos y conexiones, también se dispone de los metadatos (o datos sobre los datos) generados durante la transformación anterior.

- *Datos a Visualizar*: Este estado representa un subconjunto de los Datos Abstractos. Está constituido por *todos* los datos que van a ser visualizados de alguna manera. Es un estado que se justifica claramente desde lo conceptual.

Es importante tener en cuenta que para un mismo conjunto de Datos Crudos se pueden tener entonces más de un conjunto de Datos a Visualizar. Estos datos pueden no ser aún directamente visualizables.

- *Datos Mapeados Visualmente*: Este es el estado previo a la generación de la representación visual. Es el resultado de mapear visualmente los Datos a Visualizar. Los datos contenidos en este estado son directamente visualizables aplicando una técnica que los soporte.

La característica distintiva de este estado es la presencia de un sustrato espacial que directamente pudo haber estado ausente en el estado anterior, precisamente ésta es una de las diferencias básicas entre Visualización de Información y Científica.

$$\text{Datos Mapeados Visualmente} = \text{Datos a Visualizar} + \text{Estructura Visual}$$

donde la Estructura Visual está determinada por:

$$\begin{aligned} \text{Estructura Visual} = & \text{Sustrato Espacial} + \text{Elementos Visuales} \\ & + \text{Atributos gráficos de los elementos visuales} \end{aligned}$$

En este punto resulta evidente que para un determinado conjunto de Datos a Visualizar pueden coexistir varios conjuntos de Datos Mapeados Visualmente. Esta característica es relevante en el marco de un proceso de exploración del espacio de datos, donde el usuario puede o necesita comparar y analizar distintas maneras de visualizar un mismo conjunto de datos.

- *Datos Visualizados*: Es el último estado del proceso de visualización. No significa que sea el estado final del proceso cognitivo que está llevando a cabo el usuario, todo lo contrario, constituyen su espacio de exploración. Es muy probable a partir de una visualización el usuario comience a interactuar para lograr el *insight de los datos* que es el objetivo buscado.

### 3.2 Las transformaciones

Son los procesos que permiten pasar de un estado al próximo, en todos los casos son procesos que forman parte de una aplicación interactiva pudiendo por lo tanto requerir interacción del usuario.

Como se desprende de la Fig. 2, vemos que en nuestro modelo distinguimos cuatro transformaciones:

- *Transformación de Datos Crudos a Datos Abstractos*:

Esta la transformación permite transformar los Datos Crudos en Datos Abstractos. Es decir a partir de las fuentes de datos, externas o provenientes de otras visualizaciones, seleccionadas por el usuario deberá generar los Datos Abstractos. Esta transformación puede generar un nuevo conjunto de Datos Abstractos o directamente incorporarlos a un conjunto existente según lo indique el usuario.

- *Transformación de Datos Abstractos a Datos a Visualizar*:

De todos los datos y relaciones seleccionados por el usuario para la generación de los Datos Abstractos es frecuente que éste sólo quiera visualizar un subconjunto. Esta transformación permite realizar ese trabajo de selección, proyección y filtrado. En función de lo expresado por el usuario se generan nuevos conjuntos pertenecientes al estado de Datos a Visualizar, manteniendo para todos ellos el mismo conjunto de Datos Abstractos. El objetivo último de esta transformación es permitir que el usuario defina qué es lo quiere visualizar en una determinada instancia, sin preocuparse todavía del cómo.

- *Transformación de Mapeo de Visualización*:

Es una de las transformaciones clave en todo el proceso de visualización. Es una transformación netamente interactiva, y permite definir al usuario cómo quiere visualizar sus datos.

Es la transformación encargada de realizar el mapeo visual, es decir se deben establecer:

- ▶ Cuáles son las estructuras visuales adecuadas.
- ▶ Qué atributos se mapearán espacialmente y cómo.
- ▶ Qué elementos visuales se utilizarán y con qué atributos gráficos.

El objetivo último de esta transformación es lograr un mapeo expresivo y efectivo. Decimos que un mapeo es expresivo cuando son representados todos y absolutamente todos los datos pertenecientes al conjunto de Datos a Visualizar. Por otro lado, la efectividad del mapeo estará dada por la manera en que la representación visual sea percibida por el usuario.

Como puede intuirse, esta transformación no es trivial y gran parte de su éxito depende del conocimiento que se le imprima de las características perceptuales del humano. Es particularmente debido a lo expresado que el usuario debe explorar y comparar distintas alternativas de representación hasta lograr los resultados buscados.

- *Transformación de Visualización:*

Es la transformación encargada de generar la representación visual en pantalla según lo expresado en los Datos Mapeados Visualmente. Para un determinado conjunto de Datos Mapeados Visualmente pueden existir varias técnicas que lo soporten, debiendo el usuario optar por alguna de ellas para obtener de esta manera los Datos Visualizados.

### **3.3 La interacción**

Se ha expresado reiteradamente que la visualización debe ser un proceso interactivo. La visualización es un proceso cognitivo mediante el cual iterativa e interactivamente el usuario explora el espacio de sus datos.

Es claro que la interacción puede ocurrir en distintos puntos a lo largo de todo el proceso de visualización. Una vez realizada la primer iteración, es decir una vez que se obtuvo la primer representación visual, el usuario interactúa en la última etapa del proceso. El hecho de que las interacciones tengan lugar en el estado de los Datos Visualizados no quiere decir que se puedan resolver en ese estado.

La interacción está reflejada, sin duda en el modelo a nivel de los distintos estados y las correspondientes transformaciones.

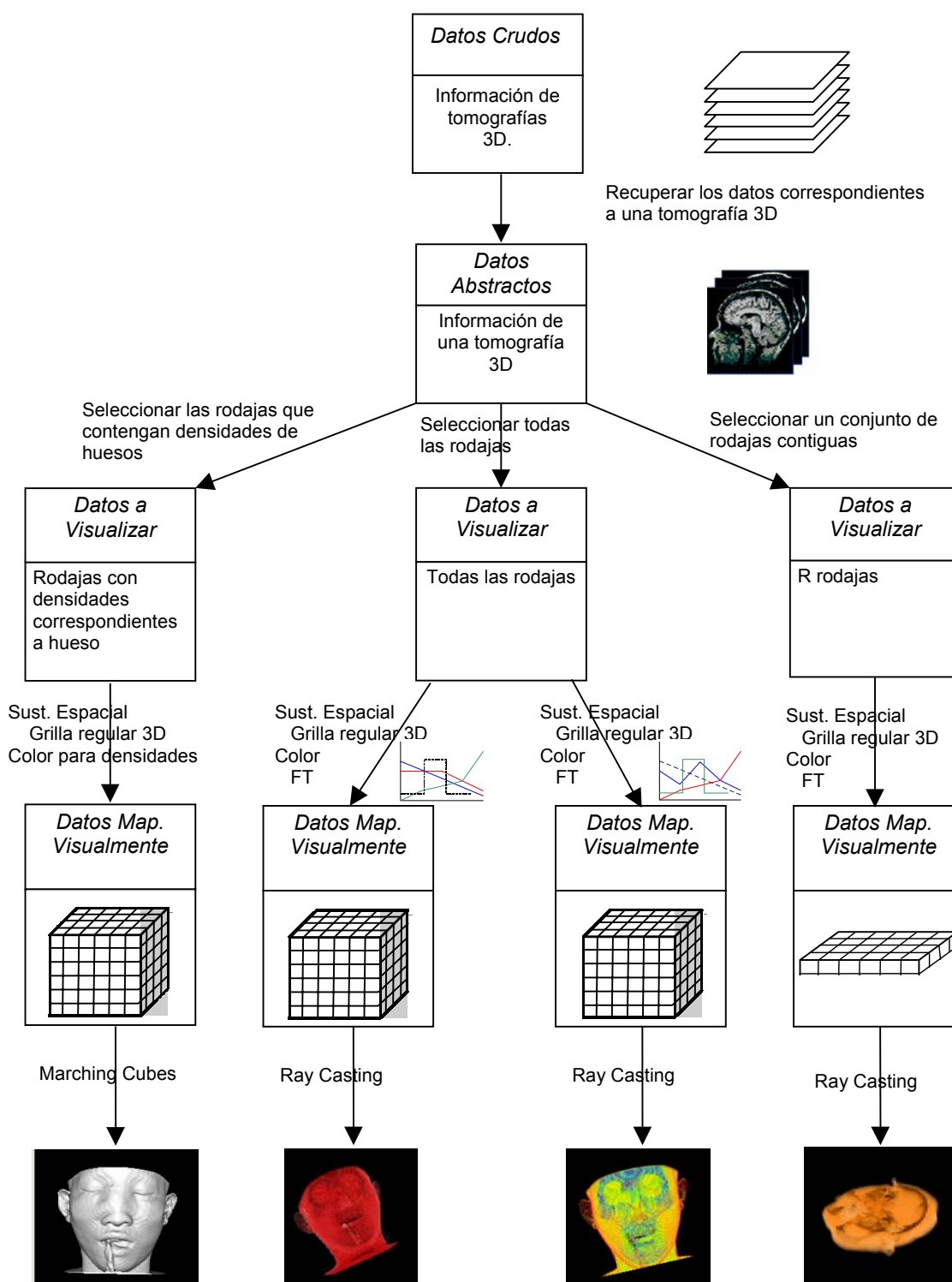
## **4 Ejemplos**

Para mostrar cómo el modelo soporta aplicaciones provenientes de las distintas ramas de visualización elegimos distintos ejemplos representativos y los ubicamos dentro del modelo. Para cada aplicación, partiendo de los datos crudos, se van aplicando las sucesivas transformaciones generando los sucesivos estados intermedios hasta alcanzar finalmente las representaciones visuales correspondientes.

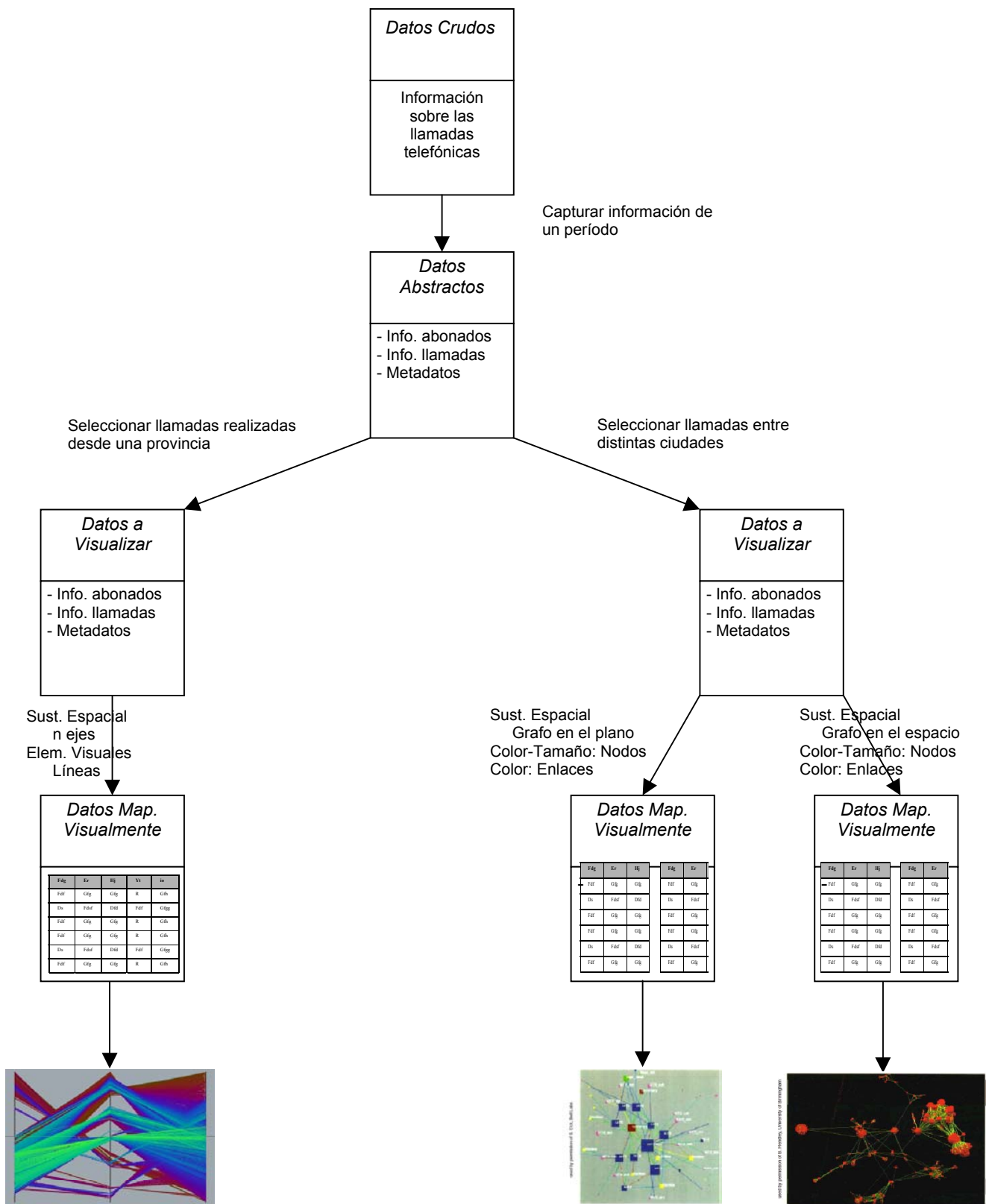
En el ejemplo de Visualización Científica que se muestra (Fig. 3), se muestran distintas visualizaciones posibles de acuerdo a las elecciones hechas en cada etapa del proceso. En este caso, se supone que se cuenta con un conjunto de tomografías y se elige una de éstas para ser visualizada. De acuerdo a distintas necesidades, se puede querer ver el conjunto 3D de una determinada tomografía o sólo parte de ésta. Cuando se trata de visualizar esta última opción puede querer visualizarse sólo un tipo de tejido (como el óseo en este ejemplo) o una parte específica del conjunto volumétrico.

En el ejemplo de Visualización de Información (Fig. 4) también se muestran distintas visualizaciones posibles. En este caso, el objetivo es visualizar distintos aspectos de información sobre las llamadas telefónicas realizadas por los abonados a una empresa de telefonía.





**Fig. 3: El modelo en Visualización Científica**



**Fig. 4: El modelo en Visualización de Información**

## 5 Conclusiones y trabajo futuro

La exploración visual de distintos espacios de información se ha convertido en uno de los tópicos más candentes de investigación en los últimos años. Esto ha conducido a que se desarrollasen distintos paradigmas de visualización y se encuadrasen en distintos marcos; por otro lado se deben considerar las distintas ramas de visualización.

En este contexto se plantea tanto para el diseñador de la visualización como para el usuario de la misma la necesidad de contar con un único marco conceptual consistente con sus intenciones. En este marco debe estar claro cuáles son las transformaciones que deben sufrir los datos, cuál es el conjunto básico de operaciones que se deben proveer y la manera en que interactúan. El modelo los debe asistir en la evaluación y ejecución de las tareas adecuadas para alcanzar la meta propuesta.

En este trabajo se ha presentado un Modelo Unificado de Visualización que apunta esencialmente a brindar un único marco conceptual de manera independiente del dominio de aplicación de origen. En este modelo se describen las transformaciones y los estados intermedios de los datos, lo que determina que se pueda enfocar sobre los operadores o sobre los operandos favoreciendo así el entendimiento de interacciones posibles.

Por otro lado, el flujo de transformaciones propuesto y los sucesivos estados que van asumiendo los datos permiten reflejar los resultados propios de las interacciones de los usuarios. El modelo soporta la existencia de varios conjuntos de Datos a Visualizar generados a partir de un único conjunto de Datos Abstractos. Los conjuntos de Datos a Visualizar pueden ser mapeados de diversas maneras determinando la existencia de los correspondientes conjuntos de Datos Mapeados Visualmente. Cada Conjunto de Datos Mapeados Visualmente puede ser visualizado aplicando más de una técnica que lo soporte y generando en consecuencia los correspondientes Conjuntos de Datos Visualizados. Es decir que el usuario dispone y puede manipular las distintas abstracciones para realizar su tarea de análisis, comparación y comprensión del espacio de información visualizado que es el fin último de la aplicación.

Actualmente y basándonos en este modelo estamos trabajando en la definición concreta de las operaciones involucradas en cada etapa del proceso de visualización y su impacto en las restantes. Es muy importante caracterizar las distintas operaciones que abraquen las distintas técnicas de visualización aplicables a las distintas ramas de visualización; esto nos permitirá detallar una arquitectura general que será el punto de partida de nuestro sistema de visualización.

## Bibliografía

1. Baecker, R. M. and Buxton, W. A. S., *Readings in Human-Computer Interaction*, San Mateo CA., Morgan Kaufmann Publishers, 1995.
2. Baeza-Yates, R., Ribeiro-Neto, B., *Modern Information Retrieval*, Addison Wesley, 1999.
3. Bass, L.; Clements, P.; Kazman, R., *Software Architecture in Practice*, Addison Wesley, 1998.
4. Bosch, J., *Design & Use of Software Architectures: Adopting and Evolving a product-line approach*, Addison Wesley, 2000.
5. Card, S., Mackinlay, J., *The structure of the information visualization design space*, Proceedings of the Symposium on Information Visualization '97, IEEE CS, 1997.
6. Card, S., Mackinlay, J., Shneiderman, B., *Readings in Information Visualization - Using Vision to Think*, Morgan Kaufmann, 1999.
7. Chi, Ed H., Riedl, J., *An operator interaction framework for visualization systems*, Proceedings of the Symposium on Information Visualization '98, IEEE CS, 1998.

8. Chi, Ed H., *A Framework for Information Visualization Spreadsheets*, Ph.D. Thesis, University of Minnesota, Computer Science Department, March, 1999.
9. Derthick, M., Roth, S., *Data Exploration across Temporal Contexts*, Proceedings of Intelligent User Interfaces (IUI'00), pp. 60-67, New Orleans, January 2000.
10. Dikel, D.; Kane, D., Wilson, J., *Software Architecture. Organizational Principles and Patterns*, Prentice Hall Inc., 2001.
11. Dix, A., Finlay, J., Abowd, G., Beale, R., *Human-Computer Interaction*, Prentice Hall Europe, Second Edition, 1998.
12. Eick, S., *Visual Discovery and Analysis*, IEEE Transactions on Visualization and Computer Graphics, Vol. 6, No. 1, January-March 2000.
13. Godfrey, P., Grant, J., Gryz, J., Minker, J., *Integrity Constraints: Semantics and Applications*, pp. 265-306. In Chomicki, J., Saake, G., *Logics for Databases and Information Systems*, pp. 31-70, Kluwer Academic Publishers, 1998.
14. Mayhew, D., *The Usability Engineering Lifecycle*, Morgan Kaufmann, 1999.
15. Mc Clure, C., *Software Reuse Techniques. Adding Reuse to the Systems Development Process*, Prentice Hall Inc., 1999.
16. Meyer, J., *Dynamic Logic for Reasoning about Actions and Change*, pp. 281-314. In Minker, J., *Logic Based Artificial Intelligence*, Kluwer Academic Publishers, 2000.
17. Nielsen, J., *Usability Engineering*, Morgan Kaufmann, 1993.
18. Nielson, G., Hagen, H., Müller, H., *Scientific Visualization: Overviews, Methodologies and Techniques*, IEEE Computer Society, 1997.
19. Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., Van de Velde, Wielinga, B., *Knowledge Engineering and Management, The Common KADS Methodology*. MIT Press, 2000.
20. Schroeder, W., Martin, K., Lorensen, B., *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, Prentice Hall PTR, 1996
21. Shneiderman, B., *Designing the User Interface*, Addison-Wesley Publishing Company, 1998.
22. Spence, R., *Information Visualization*, Addison-Wesley, ACM Press, 2001.
23. Stasko, J., Domingue, J., Brown, M., Price, B. editores. *Software Visualization: Programming as a Multimedia Experience*, MIT Press, 1998.
24. Tufte, E., *The Visual Display of Quantitative Information*, Graphics Press, 1983.
25. Tufte, E.R., *Envisioning Information*, Cheshire, CT Graphics Press, 1990.
26. Tufte, E.R., *Visual Explanations: Images and Quantities, Evidence and Narrative*, Cheshire, CT Graphics Press, 1997.