

207
207



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO.

FACULTAD DE INGENIERÍA

BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA. UN MODELO BÁSICO.

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN.
P R E S E N T A :
GUSTAVO CORNEJO GONZÁLEZ.

DIRECTOR:
MAESTRO ALFONSO MIGUEL REYES.

CODIRECTOR:
ING. SERGIO AMBRIZ MAGUEY.



MÉXICO, D.F.

1998

TESIS CON
FALLA DE ORIGEN

263689



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

BROWSER GRÁFICO PARA LA
VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA.
UN MODELO BÁSICO.

DEDICO LA PRESENTE TESIS:

A la memoria de mi Padre.

Agustín Cornejo López.

A mi Madre Guadalupe González Hernández por todo el apoyo, cariño y comprensión que me ha brindado incondicionalmente y por ser la luz que ilumina mi vida día con día.

A todos mis hermanos y hermanas.

A mi director de tesis Alfonso Miguel Reyes y al Instituto Mexicano del Petróleo (IMP) por el apoyo brindado para realizar este trabajo.

A mi codirector Sergio Ambriz Maguey por todo el tiempo, paciencia y consejos.

A mis amigos y amigas que compartieron conmigo momentos maravillosos a lo largo de la carrera y por demostrarme siempre su lealtad y honestidad.

A todos mis compañeros del IMP.

Y muy especialmente a la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería por darme la invaluable oportunidad de estudiar en sus aulas.

ÍNDICE

Capítulo 1. Introducción.....	1
Capítulo 2. Taxonomía de los browsers	
2.1. ¿Qué es un browser?.....	4
2.2. ¿Dónde se utiliza un browser?.....	4
2.3. Browsers de dos dimensiones vs. Browsers de una dimensión.....	5
2.4. ¿Cómo se pueden determinar las características de un browser?.....	5
2.5. Notación empleada en la especificación de un browser.....	6
2.5.1. Restricción de movimiento.....	6
2.5.2. Restricción proporcional del tamaño.....	6
2.5.3. Campo visual.....	6
2.5.4. Capacidad de proyección.....	7
2.5.5. Coordinación entre ventanas.....	8
2.5.6. Especificación del browser.....	8
2.6. Comandos.....	8
2.7. Clasificación de los browsers.....	9
2.7.1. Browser de un solo detalle.....	9
2.7.2. Simple ventana con zoom y recemplazo.....	10
2.7.3. Simple par de ventanas coordinadas.....	11
2.7.4. Múltiples niveles adosados.....	12
2.7.5. Múltiples superposiciones y libre zoom.....	12
2.7.6. Vista bifocal.....	13
2.7.7. Vista de ángulo plano.....	14
2.8. Tareas de un browser.....	14
2.9. Aspectos de presentación de un browser.....	15
2.9.1. Estáticos.....	15
2.9.1.1. Browser de simple vista.....	15
2.9.1.2. Browser de múltiples vistas.....	16
2.9.2. Dinámicos.....	17
2.9.2.1. Calidad de la actualización.....	17
2.9.2.2. Naturaleza de la actualización.....	17
2.9.2.3. Factor de zoom.....	17
2.10. Operaciones de un browser.....	17
2.10.1. Operaciones manuales.....	17
2.10.1.1. Zooming.....	18
2.10.1.2. Panning.....	18
2.10.2. Operaciones automáticas.....	18
2.10.2.1. Salvar puntos.....	18
2.10.2.2. Navegación.....	19
2.10.2.3. Manejo de ventanas.....	19
2.10.2.4. Bosquejo de la imagen.....	19

Capítulo 3. Características de la información petrolera.

3.1.Historia de los registros de pozo.....	20
3.2.Obtención de los registros.....	21
3.3.Almacenamiento de los registros geofísicos de pozo.....	23
3.4.Contenido de un registro.....	25
3.5.Antecedentes del sistema VIPREG.....	28

Capítulo 4. Análisis y Diseño del Browser

4.1.Análisis.....	30
4.1.1.Definición del problema.....	30
4.1.2.Objetivo.....	30
4.1.3.Metodología empleada.....	31
4.1.4.Requerimientos del Browser para la Visualización de la Información Petrolera (VIPREG).....	33
4.1.5.Elementos de la gráfica de datos.....	36
4.1.6.Características generales de la gráfica.....	37
4.1.7.Atributos de las curvas.....	38
4.1.8.Rangos de carga de operación estimados del sistema.....	39
4.1.9.Análisis composicional.....	39
4.1.10.Conclusiones de carga de operación del sistema.....	39
4.1.11.Modelo inicial del sistema(diagrama de alto nivel).....	41
4.1.12. Especificación preliminar de los componentes.....	42
4.1.13.Análisis Ambiental.....	43
4.1.14.Análisis funcional interno (diagramas de trazas).....	66
4.2.Diseño del sistema.....	83
4.2.1.Características generales de diseño.....	83
4.2.2.Ambiente de desarrollo.....	84
4.2.3.Modelo inicial del sistema.....	85
4.2.4.Diseño funcional.....	88
4.2.5.Diagramas de interacción.....	88

Capítulo 5. Construcción del modelo básico

5.1.Especificación del Browser.....	107
5.2.MENÚ PRINCIPAL.....	110
5.3.CARACTERIZADOR.....	114
5.4.CURVÍGRAFO.....	119
5.5.VISOR.....	124
5.6.DIÁLOGO PARA MODIFICAR LAS PROPIEDADES DE LA GRÁFICA.....	132
5.7.DIÁLOGO PARA MODIFICAR LOS ATRIBUTOS DE LAS CURVAS.....	134

Capítulo 6. Pruebas.....136

Capítulo 7. Conclusiones.....141

Capítulo 8. Bibliografía.....143

1. INTRODUCCIÓN

Los yacimientos petroleros se definen como unidades de acumulación de hidrocarburos, representados por un volumen continuo de terreno impregnado.

Para su localización y explotación requieren básicamente de dos tipos de información: Registros Geofísicos de Pozo e Información sísmica; los cuales proporcionan los elementos necesarios para su ubicación tanto espacial como temporal de la secuencia representada por ellos.

Los Registros Geofísicos proporcionan la información necesaria para la identificación de propiedades petrofísicas de las diferentes formaciones que un pozo atraviesa, por medio de análisis cualitativos y cuantitativos, para lo que son requeridas herramientas que permitan realizarlos. Entre estas se encuentran visualizadores o browsers que proporcionen el despliegue/visualización de la información de Registros Geofísicos en forma gráfica, debido principalmente al gran volumen de datos. Los visualizadores ofrecen características a los distintos usuarios para la realización de análisis cualitativos o de apoyo en las diferentes operaciones de la Industria Petrolera como son : análisis de registros para evaluaciones litológicas y petrofísicas; correlaciones pozo a pozo, análisis texturales y estructurales de paredes de pozo, etc.

Actualmente los principales visualizadores existentes son de procedencia extranjera, de alto costo de adquisición y mantenimiento, además no incluyen características propias de las operaciones de la Industria Nacional, y la estructura organizacional de PEMEX dividida en diferentes activos (unidades de trabajo definido por uno o varios campos petroleros), hace necesarias el desarrollo de un visualizador que pueda ser utilizado en los diferentes equipos de la industria.

El objetivo del presente trabajo es presentar el desarrollo de un visualizador gráfico (browser) en el contexto de la Industria Petrolera empleando técnicas de Ingeniería de Software con orientación a objetos.

El capítulo 2 describe las principales características de los browsers, su clasificación, tareas y funcionamiento, como un contexto general dentro del cual se puede especificar el browser. El propósito del capítulo es presentar las alternativas para determinar las características que más se adecuen al tipo de información que se maneja dentro de la Industria Petrolera.

El capítulo 3 proporciona las características básicas de la información petrolera, especialmente en el área de Registros Geofísicos de Pozo, con el fin de establecer la importancia de la construcción de un browser que maneje esta información. Además de proporcionar al lector los conceptos básicos para comprender el ambiente en el que trabajará el browser.

El capítulo 4 presenta los requerimientos del browser especializado que se maneja en el IMP y los centros de cómputo para los cuales estará destinado el sistema, con el fin de establecer los componentes y las características de cada uno de éstos. Una vez determinados los requerimientos y los componentes por medio del análisis, es posible realizar el diseño del browser de acuerdo a los métodos y/o alternativas que presenten una mejor solución a nuestro sistema. En este trabajo se optó por una metodología denominada Diseño de Software Orientado a Objetos con Calidad (DSOOC), que es un estándar que se maneja en la Línea de Negocio y Adquisición de Datos de Pozo.

Además se presenta el desarrollo del análisis y diseño empleando la metodología antes mencionada, proponiendo un modelo del browser a partir del cual se construirá un modelo básico en el capítulo 5. La implementación es en un lenguaje novedoso (Builder C++), que facilita en gran parte la programación de sistemas bajo un ambiente Windows.

El capítulo 6 presenta las pruebas necesarias realizadas para determinar la eficiencia, validación y la calidad del browser. Las pruebas determinarán el nivel de confiabilidad del sistema.

Por último el capítulo 7 describe las conclusiones que testifican los objetivos de la presente Tesis, permitiendo al mismo tiempo comprobar la validez, utilidad y la eficiencia del sistema desarrollado para la Industria Petrolera.

2. TAXONOMÍA DE LOS BROWSERS¹

Este capítulo introduce al lector en el contexto del concepto de *browser*. ¿Para qué se utiliza?. ¿Por qué del empleo de un *browser*?, las características, clasificación y operaciones que se realizan con ellos. También se incluyen algunas guías y consejos para su diseño.

2.1 ¿Qué es un browser?

La mayoría de las aplicaciones que requieren visualizar una imagen, gráfica o documento más grande que una pantalla, tiene la necesidad de un elemento capaz de manejar este volumen de información, el cual es conocido con el nombre de *browser*¹ (hojeador).

Así, los browsers han tenido mucha aceptación en este tipo de aplicaciones, ya que representan una herramienta de gran utilidad para realizar el recorrido, análisis visualización y edición de imágenes, gráficas, documentos, etc.

2.2 ¿Dónde se utiliza un browser?

- **En programas de pintar y dibujar.** En este tipo de aplicaciones, la imagen sobrepasa el tamaño de la pantalla, por lo general el tiempo de despliegue es largo y surge la necesidad de emplear acercamientos de la imagen, resultando indispensable el uso de “*scroll bars*” para realizar el recorrido de la imagen. Cabe destacar que sin el empleo de éstos, el recorrido de la imagen/gráfica tendría que llevarse a cabo por medio de comandos que complican al usuario la utilización del programa.

Por ejemplo, en el recorrido de un documento de una posición a otra se usaría el comando “100,150,despliega”, entonces el programa desplegaría de la línea 100 a la 150; en cambio, por medio del “*scroll bar*”, el usuario sólo lo desliza hasta la posición o líneas que desea desplegar, sin la necesidad de recordar la posición en la cual escribió cierta información.

- **En sistemas de información geográfica.** Para la visualización y recorrido de un mapa mundial, es necesario realizar acercamientos a un país, ciudad o localidad, utilizando un *browser* es posible visualizar una vista global, (un mapa mundial por ejemplo), una vista intermedia (Europa o Francia), una vista detallada/local (París), la cual muestra usualmente una parte de la imagen magnificada, etc.
- **En Medicina.** Un doctor puede requerir ver en el monitor de una computadora una columna vertebral completa y después enfocarse sobre algunos pares de vértebras en particular, y/o examinar los tejidos. Los browsers son de gran utilidad ya que permiten el manejo de estas imágenes y los doctores pueden diagnosticar con cierta exactitud las anomalías presentadas, por medio del análisis de las imágenes presentadas en la pantalla de la computadora.

- **En aplicaciones como distribución de poder, redes telefónicas, sistemas de administración y plantas químicas**, donde se manejan típicamente diagramas sobre el monitor de una computadora, y el usuario puede detectar a través del estudio de éstas imágenes ciertas anomalías presentadas en el sistema o emplear vistas detalladas por medio de acercamientos para dictaminar con más certeza el problema presentado.
- **En la industria petrolera**. Dentro de la industria petrolera existe información conocida como registros geofísicos de pozo. Un registro es presentado por medio de una gráfica de propiedades físicas de la roca (información tomada por medio de sondas y almacenada en forma de columnas -archivo LAS-) contra la profundidad. Por lo general la imagen/gráfica resultante es muy grande (de 8 o más pantallas).

2.3 Browsers de dos dimensiones vs. Browsers de una dimensión

Los browsers de una dimensión son aquellos que pueden ser explorados en una sola dirección, por ejemplo un editor de texto en el que la achura del documento permanece constante y no mayor del ancho de la pantalla, y solo es empleado un scroll bar vertical para realizar su recorrido.

Los browser de dos dimensiones son aquellos que pueden explorar en ambas direcciones horizontal y verticalmente. Algunas veces se pretende simular browsers de dos dimensiones con dos scroll bar de una dimensión, lo cual no siempre es eficiente, ya que depende de la aplicación dentro de la cual trabaja. Los *scroll bar* de una dimensión tienen un funcionalidad en común, pero sus características individuales y de operación difieren substancialmente, por lo que solo resultan efectivos si la exploración se realiza en una sola dirección por pequeños incrementos menores que una pantalla.

El análisis y diseño de los browser de dos o más dimensiones requieren de un estudio más cuidadoso para obtener mejoras significativas. En la siguiente sección se describe una especificación técnica informal para describir los browsers de 2D y una clasificación de acuerdo a su presentación, características, tareas, etc. También se sugieren consejos y estrategias existentes para su diseño.

2.4 ¿Cómo se pueden determinar las características del browser?

En muchas ocasiones, los diseñadores de browsers tienen dificultades para determinar las características que debe poseer un browser para una aplicación específica. En la Universidad de Maryland se desarrolló DMSketch (Manipulación directa del esbozo), el cual auxilia a los diseñadores a cambiar y guardar ideas más rápida y claramente que una especificación formal del lenguaje. Además DMSketch muestra una mejor diferenciación de las características del browser (elementos mas significativos de la gráfica, la relación que existe entre ellos y las acciones más importantes que puede realizar el usuario).

DMSketch esta basado en una técnica de Scott Hudson y Shamin Mohamed⁴ la cual emplea la representación gráfica de las interfaces del usuario.

2.5 NOTACIÓN (Primitivas empleadas en DMSketch)

2.5.1 Restricción de movimiento

Especifica que el objeto es movable, en dirección horizontal si la flecha es horizontal y en dirección vertical si la flecha es vertical. Sin la existencia del operador de movimiento, el objeto no es movable. Fig. 2.1.

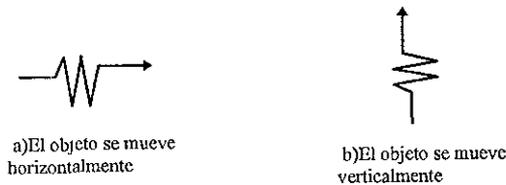


Fig. 2.1. Símbolo para representar una restricción de movimiento. a) Horizontal, b) vertical.

2.5.2 Restricción proporcional del tamaño

Este símbolo une a dos objetos por los puntos del círculo final, esta primitiva fuerza a dos objetos unidos a mantener el mismo tamaño relativo. Fig. 2.2

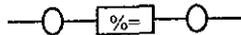


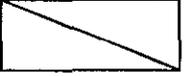
Fig. 2.2. Símbolo para representar una restricción proporcional del tamaño entre dos objetos

2.5.3 Campo Visual

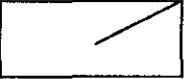
Encierra una parte de una imagen regularmente delimitado por un rectángulo, que es proyectado sobre una nueva ventana y generalmente magnificado, lo cual es definido por una flecha punteada desde el campo visual hasta la ventana destino. A continuación se muestran 5 casos de la representación de un campo visual. Fig. 2.3



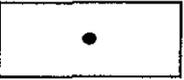
Generalmente representado por las acciones “*Mouse down, drag y Mouse up*”, este tipo de campo visual es variable y sus dimensiones dependen de dichas acciones.



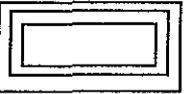
El campo visual es fijo y determinado por una esquina.



El campo visual es definido por un punto en el centro y cuya longitud varía de acuerdo a la distancia con las esquinas.



El campo visual es definido por un punto en el centro y siempre es del mismo tamaño.



Representa un campo visual con varias ampliaciones disponibles.



Campo visual en forma de “bota”, misma forma que toma la ventana destino.

Fig. 2.3 Diversas formas de representar un campo visual.

2.5.4 Capacidad de proyección

Indica que la imagen dentro del campo visual es proyectada sobre una nueva ventana.
Fig. 2.4

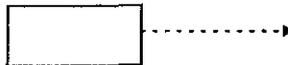


Fig. 2.4. Símbolo para representar que la región seleccionada por el rectángulo es proyectada a una nueva ventana

2.5.5 Coordinación estándar entre ventanas

Indica que la actualización o movimiento en una ventana también afecta la otra ventana, en otras palabras, existe una sincronización entre ventanas. Fig. 2.5.

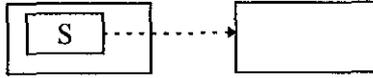


Fig. 2.5 Símbolo para indicar que existe coordinación entre ventanas

2.5.6 Especificación del Browser

Para simplificar la especificación pueden emplearse objetos compuestos, los cuales toman un símbolo propio y pueden ser empleados en especificaciones subsecuentes. Fig. 2.6.

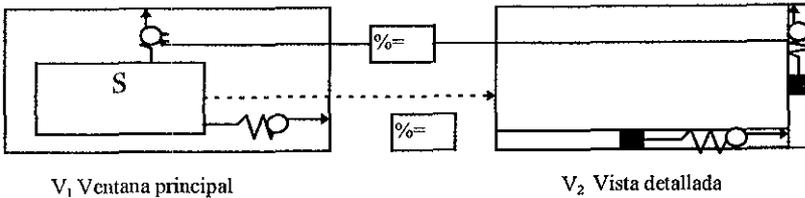


Fig. 2.6. Ejemplo de la especificación de un browser.

La ventana izquierda V₁ es la vista principal, el campo visual S puede moverse horizontal y verticalmente dentro de V₁, la parte de la imagen encerrada por el campo visual es proyectada a una segunda ventana V₂ - vista detallada -, la cual contiene scroll bars ligados y sincronizados directamente al campo visual dentro de la ventana principal. Por medio de esta coordinación entre ventanas, moviendo la posición del campo visual en la ventana principal cambiamos la parte desplegada de la imagen dentro de la vista detallada V₂; al mismo tiempo se actualizan las posiciones de los scroll bars, y viceversa, al cambiar la posición de los scroll bars en la vista detallada, automáticamente movemos la posición del campo visual S dentro de la ventana principal.

2.6 COMANDOS

No existe un estándar para describir los comandos que puede contener un browser. DMSketch emplea la fig. 2.7 para mostrar el estado antes y después de la interface, los cuales se presentan con el nombre del comando asociado con una flecha direccionada.

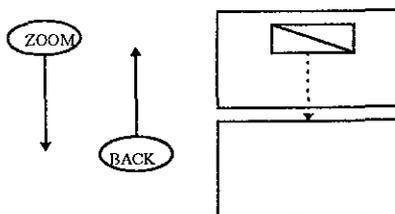


Fig. 2.7. Notación para señalar una operación de zoom

2.7 CLASIFICACIÓN DE LOS BROWSERS

Existe gran diversidad para el diseño de un browser, las características de este dependen de las necesidades del sistema, debido a que lo que funciona correctamente en un sistema, puede fallar en otro. A continuación se enlistan las principales formas en que puede trabajar un browser.

2.7.1 Browser de un solo detalle

La imagen/gráfica es presentada como una sola ventana que puede ser recorrida horizontal y verticalmente sobre la vista detallada de la imagen, pero con el inconveniente de que solo es útil si el factor de zoom es relativamente pequeño y la vista global no siempre tiene que permanecer visible. Fig. 2.8

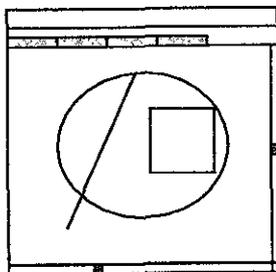
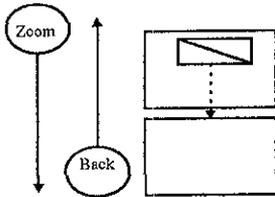


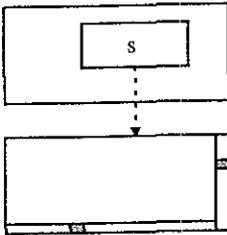
Fig. 2.8 Ejemplo de una aplicación donde solo existe una vista en la cual es presentada la imagen/gráfica.

2.7.2 Simple Ventana con zoom y reemplazo

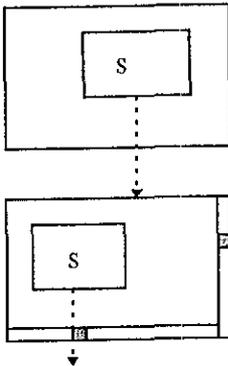
Esta técnica presenta una vista global de la imagen. Dentro de la vista global se puede marcar un rectángulo -campo visual- conteniendo una parte de la imagen, la cual es magnificada y reemplazada dentro de la misma ventana donde se encontraba la imagen global. Es relativamente fácil de implementar y emplea eficientemente el espacio de la ventana. Existen tres casos para este tipo de browser:



1. Tiene el inconveniente de que solo magnifica el espacio encerrado en el rectángulo y cuando se desea ver otra parte de la imagen es necesario regresar a la vista global.



2. Magnifica toda la imagen y en la ventana se reemplaza la parte de la imagen encerrada en el rectángulo, el usuario puede seguir examinando el resto de la imagen por medio del desplazamiento de los scroll bars de la vista detallada.

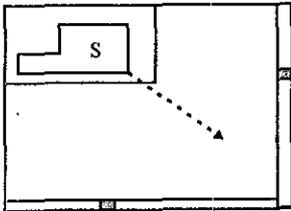


3. Magnifica la imagen encerrada en el rectángulo, reemplazándola en la ventana y colocando scroll bars. Dentro de la nueva presentación de la imagen (primera magnificación) puede seleccionarse un nuevo parte de la imagen por medio de otro rectángulo y volver a magnificarlo y reemplazarlo dentro de la misma ventana, este procedimiento es conocido como múltiples niveles de magnificación.

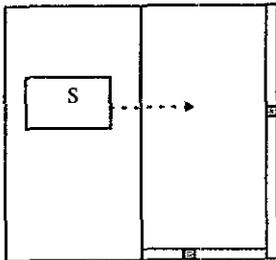
Fig. 2.9 Diversos tipos para representar un browser de una simple ventana con zoom y reemplazo
Los tres tipos antes descritos pueden retornar a la vista global de la imagen

2.7.3 Simple par de ventanas coordinadas

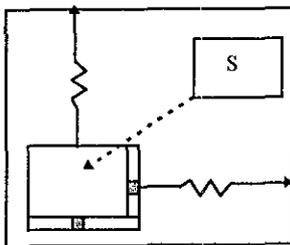
Este tipo de browsers coordinan dos ventanas, generalmente en una se encuentra la vista global y en la otra ventana una vista local magnificada. Existe una coordinación entre ambas ventanas a tal grado que si se selecciona una parte de la vista global, automáticamente es magnificada y presentada en la segunda ventana -vista local-, y por el contrario, si se realiza un desplazo en la vista local detallada automáticamente se actualiza el campo visual en la vista global. Existen tres casos para este tipo de browsers. Fig. 2.10.



1. En una esquina se presenta la vista global y en una ventana en forma de bota la vista local. Se selecciona una parte de la vista global, por medio de una forma de bota, la cual es presentada en la vista local magnificada.



2. La vista global y la vista local magnificada emplean el mismo tamaño de la pantalla. Se selecciona una parte de la vista global por medio de un rectángulo y automáticamente es presentado en la vista local magnificada. Existe coordinación entre ambas ventanas, así que lo que se realiza en una afecta a la otra.



3. En este browser, la vista global ocupa toda la pantalla y la vista detallada magnificada es presentada en una pequeña ventana móvil. De igual forma existe coordinación entre las dos ventanas.

Fig 2.10 Tipos de browsers con simple par de ventanas coordinadas

2.7.4 Browsers de múltiples niveles adosados

Este tipo de browsers combinan múltiples vistas globales, intermedias y detallados. Existe coordinación entre todas las ventanas y por lo tanto su manejo es más difícil que en otros casos. Las ventanas son presentadas en forma de mosaico de tal manera que todas son visibles y presentadas estratégicamente de acuerdo al tipo de aplicación en la que se desenvuelva. Fig. 2.11.

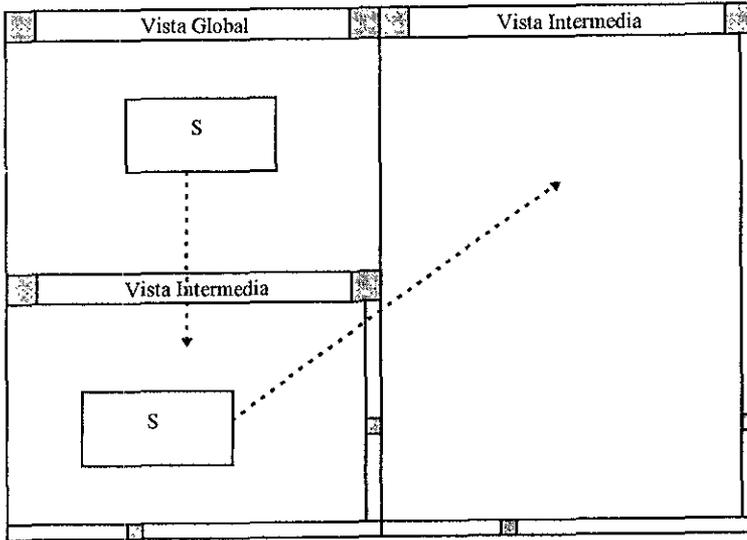


Fig 2.11. Browser con múltiples niveles adosados.

2.7.5 Múltiples superposiciones y libre zoom

Este tipo de browsers son más flexibles, pero ello se paga con la velocidad, se recomienda emplearlos en plataformas rápidas y con pantallas grandes. Los usuarios son libres de especificar, mover, rehacer y borrar las ventanas como ellos lo deseen. Por lo general la imagen completa es presentada primero, el usuario marca una parte de la imagen (la parte de la imagen seleccionada puede ser variable, el usuario determina los límites del rectángulo) y posiblemente también el factor de zoom y entonces el área seleccionada es proyectada a una nueva ventana, la cual se superpone en la ventana fuente. No existe coordinación entre las ventanas, así el usuario tiene dos browsers a niveles de magnificación diferentes.

Este tipo de browsers posee el inconveniente de que el tiempo empleado para el manejo de ventanas es considerable, debido a que las ventanas constantemente se ocultan unas de otras y en el refrescamiento de la imagen se emplean muchos recursos. La fig. 2.12 muestra a grandes rasgos el funcionamiento de este tipo de browsers, la nueva ventana creada a partir de la vista global, es considerada como un nuevo browser que puede ser manejado

independientemente de su ventana de origen, por lo que también puede tener magnificaciones.

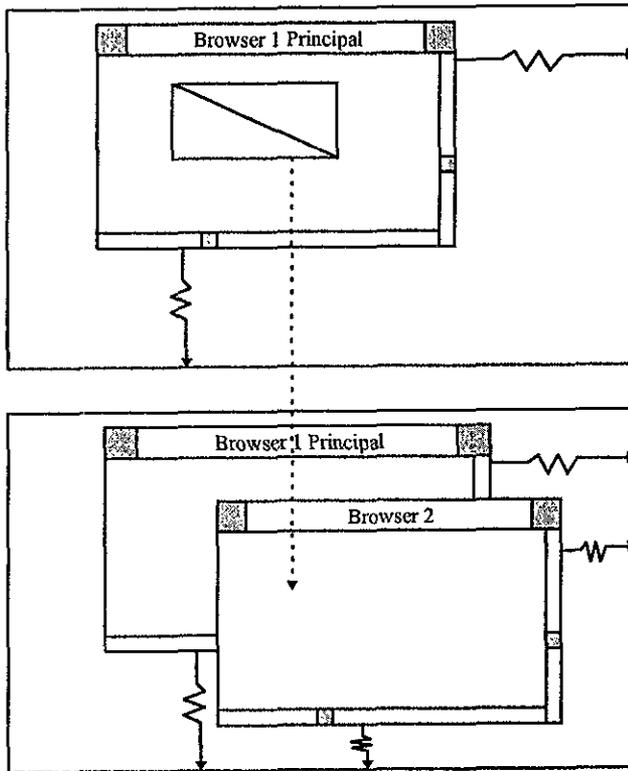


Fig. 2.12. Browser con múltiples superposiciones y libre zoom

2.7.6 Browser de vista bifocal

Es una variante de *browser* de un solo detallado clásico. Utiliza una metáfora del vidrio y los zoom de imagen son colocados sobre el área en la cual el objeto magnificado es colocado y consecuentemente con ello cubrimos los objetos vecinos.

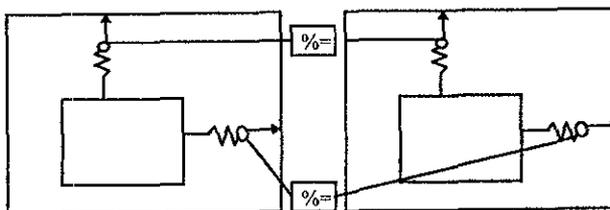


Fig. 2.13 Browser de vista bifocal.

2.7.7 Vista de ángulo plano

Es la extensión de la vista bifocal en la cual se magnifica el centro de mayor interés de la imagen de la misma ventana y el resto de la imagen es progresivamente comprimida. Se utiliza una simple vista y el *zoom* y los *scroll bars* no son requeridos, pero los usuarios deben especificar que punto desea magnificar. Poseen el inconveniente de que si la imagen es muy grande puede sufrir una distorsión severa.

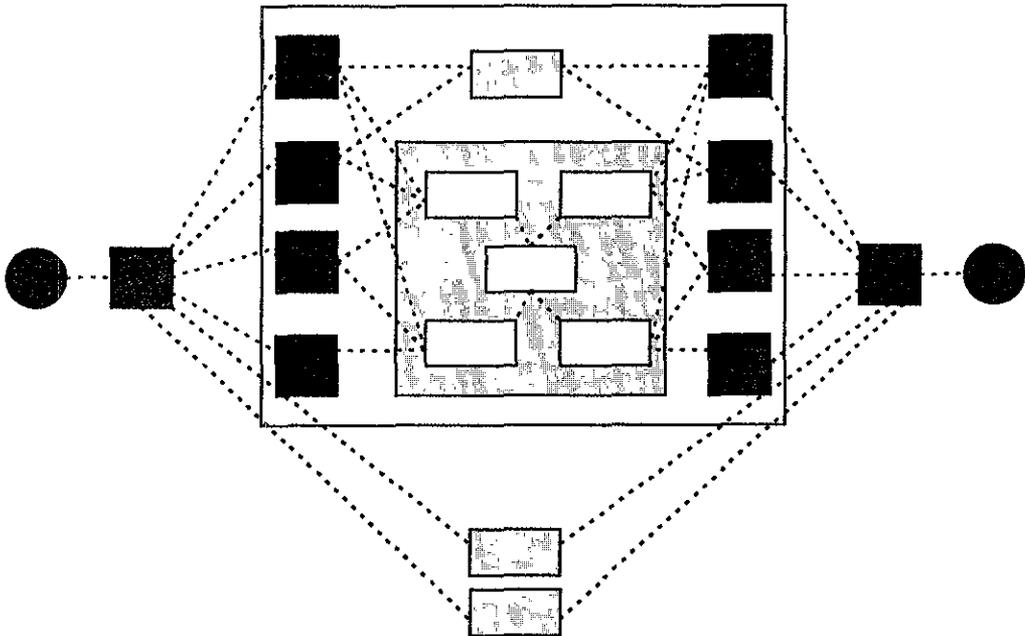


Fig. 2.14. Browser con vista de ángulo plano.

2.8 TAREAS DE UN BROWSER

Existen 5 tareas principales que debe realizar un browser.

1. **Generación de la imagen.** Consiste en la elaboración de la imagen/gráfica de tal forma que satisfaga las necesidades de la aplicación. Por ejemplo, un usuario que dibuja o pinta un diagrama largo, puede estar interesado en una parte en particular, pero tal vez después necesite regresar y observar la imagen completa.
2. **Exploración.** Se refiere a la forma en que la imagen /gráfica es examinada. Por ejemplo : en un juego de aventuras el jugador se mueve rápidamente alrededor de un espacio imaginario una y otra vez hasta que llega a ser familiar, o cuando se explora por primera vez un mapa, el espacio no es conocido por el usuario y fácilmente se puede perder, por lo tanto la exploración debe ser rápida y la interface de usuario actualizada rápidamente.

3. **Diagnóstico.** En esta tarea la velocidad de paneo debe ser rápida y el tratamiento de la imagen completa es importante, ya que por medio del diagnóstico de la imagen es posible encontrar fallas o alteraciones de la imagen/gráfica que se esta analizando. Un ejemplo de esto, puede ser la exploración por parte de un patólogo de tejido digitalizado o la exploración de la vista magnificada de un circuito VLSI por parte de un especialista.
4. **Navegación.** Se refiere al recorrido a través de la imagen/gráfica principalmente por medio de *scroll bars* o de diferentes vistas, el *zoom* ocurre ocasionalmente. La información relevante es mostrada a través de magnificaciones para poder apreciar los detalles de la gráfica.
5. **Monitoreo.** Esta tarea consiste en el análisis de la gráfica, debido a que por medio del monitoreo el usuario puede detectar problemas y asignar algunas atenciones locales para poder corregirlo. Un ejemplo de esto puede ser el manejo de una red extensa o el monitoreo de una central de seguridad. Un factor importante es el manejo de ventanas, esto debido a que una ventana superpuesta puede ocultar cambios importantes ocurridos en otra ventana.

2.9 ASPECTOS DE PRESENTACIÓN DE UN BROWSER

De acuerdo a su presentación los *browsers* se clasifican en estáticos y dinámicos.

2.9.1 Estáticos.

Este tipo de *browsers* son aquellos en los que las ventanas a través de las cuales se presenta la información permanecen constantes, es decir, no son movibles ni redimensionables. Bajo esta categoría se subclasifican en *browsers* de simple vista y *browsers* de múltiples vistas, aunque la mayoría de los *browsers* son híbridos.

2.9.1.1 Browser de simple vista

Dedican todo el espacio de la pantalla a una sola vista, son muy eficientes cuando el *paneo* es limitado y son comunmente utilizados cuando el espacio de despliegue es insuficiente. Son inapropiados cuando los usuarios deben comparar diferentes partes distantes de una imagen.

Existe tres variantes de este tipo de *browsers* :

1. **Solamente un detalle.** No soportan *zoom*, el default si la imagen es mas larga que la pantalla. Son los más comunes para la generación de imágenes debido a que la mayoría del trabajo esta listo a nivel detallado. Son inapropiados para el monitoreo.
2. **Zoom y reemplazo.** Son los más apropiados para manejar una imagen grande, la cual puede realizar acercamientos que son presentados sobre la misma ventana de la vista global. Esta gran familia de *browsers* son apropiados para la generación de imágenes y diagnósticos si la velocidad de actualización de la pantalla de despliegue es suficiente.
3. **Ángulo plano.** Da detalles del contexto en una simple vista, pero distorsiona severamente la imagen y requiere de una reorientación constante. Estos *browsers* son los más

apropiados para representaciones de vistas abstractas tales como diagramas de red. Los diseñadores deben recordar que las vistas de ángulo plano pueden ser inapropiadas cuando la fidelidad de la imagen es importante. Existen tres técnicas que modifican la vista de ángulo plano :

- La distorsión gráfica de la imagen.
- Filtros para remover objetos no conocidos desde el foco.
- Abstracción para reemplazar bloques con símbolos.

2.9.1.2 Browsers de múltiples vistas

Estos *browsers* despliegan diferentes vistas. Son empleados cuando es importante ver los detalles y el contexto de la imagen simultáneamente, cuando la distorsión del ángulo plano no es apropiado, la vista paralela es requerida para la comparación y cuando la velocidad de despliegue no es suficiente para permitir el *zoom* o *paneo* continuo. Se consideran tres tipos de *browser* de múltiples vistas.

1. **Estrategia de colocación de ventanas.** Es más fácil de implementar un *browser* de múltiples ventanas empleando una estrategia de colocación. Los recursos que se emplean para el manejo de superposición de ventanas son demasiados, así que si se diseña el *browser* de tal forma que contenga un manejo y colocación de ventanas de forma automática, el funcionamiento será más eficiente, además ayudará al usuario a no tener que mover y redimensionar ventanas innecesariamente.
2. **Coordinación.** Se considera que la coordinación entre ventanas ofrece al usuario grandes ventajas, debido a que facilita la navegación y la exploración de la imagen/gráfica y le permite observar al mismo instante diferentes magnificaciones de la gráfica sobre una misma región de forma automatizada.
3. **Vista Global.** Una vista global muestra el espacio de información completo y permite un acceso rápido a cualquier parte de la imagen. La vista global puede ser tan detallada como lo permita el *display*. Esta vista es necesaria cuando se desean visualizar imágenes/gráficas más largas que la pantalla.

Acomodando las ventanas en forma de mosaico fig. 2.11, el usuario se libera de la necesidad del manejo de éstas y por el contrario, si existe superposición de ventanas el sistema posiblemente se alentará debido al manejo de recursos y de las ventanas, pero se logra más flexibilidad del sistema. Así que todo depende de las necesidades que debe cumplir el *browser* y la plataforma en la cual trabajará.

2.9.2 Dinámicos

Bajo este tipo de browsers están los más suaves en actualización de la pantalla, el factor *zoom* es variable y las ventanas redimensionables, por lo que se desarrollan especialmente para plataformas que poseen gran velocidad para el manejo de ventanas. A continuación se describen los aspectos que debe considerar un browser dinámico para tener un mejor desempeño.

2.9.2.1 Calidad de la actualización

Un rápido, simple y continuo actualizador de la imagen hace la navegación más sencilla y natural en planos sobre distancias relativamente grandes. Así el usuario tiene mayor concentración sobre sus tareas y no se preocupa por las herramientas de navegación.

2.9.2.2 Naturaleza de la actualización

Un área seleccionada puede ser expandida o explotada para revelar su estructura interna. La explosión de la gráfica es regularmente utilizada en grupos de datos jerárquicamente seleccionados, lo cual simplifica la vista general, pero puede causar desorientación, porque la imagen siempre está cambiando. Si se emplea la expansión y/o la explosión, el *zoom* de la imagen puede ser desorientado, como en el caso del *browser* de vista de ángulo plano. Así que los diseñadores deben combinar varias técnicas. Por ejemplo, cuando el usuario selecciona un objeto, el vecindario alrededor de éste puede ser expandido y el objeto mismo explotado para mostrar su estructura interior.

2.9.2.3 Factor de zoom

Es el nivel de magnificación entre dos vistas, puede ser fijo (predeterminado por el diseñador) o variable (especificado por el usuario en tiempo de ejecución). Si el factor *r* de *zoom* es variable, la complejidad del *browser* aumenta.

2.10 OPERACIONES DE UN BROWSER

Las operaciones de un *browser* se clasifican en operaciones manuales y automáticas.

2.10.1 OPERACIONES MANUALES

Dentro de las operaciones manuales se consideran el *zoom* y el *paneo*, los cuales pueden reajustarse simultáneamente por el dibujo (actualización de la pantalla) del campo visual o ajustando el tamaño, colocación y distancia del radio.

2.10.1.1 Zooming

Los usuarios especifican manualmente la localización del *zoom* por medio de la determinación de un campo visual delimitado generalmente dentro de un rectángulo, en el cual un tamaño fijo determina un factor de *zoom* fijo y los rectángulos de tamaño variable especifican un factor de *zoom* variable. La vista magnificada o detallada puede ser colocada por el usuario o por el sistema en alguna zona predeterminada.

El diseñador del browser debe buscar un equilibrio entre complejidad y flexibilidad, debido a que entre más flexible pretenda ser un browser, mayor tendrá que ser la complejidad para su control. Por ello los diseñadores deben tener un estudio cuidadoso en las tareas que debe desempeñar un *browser*, procurando satisfacer las tareas eficientemente y al mismo tiempo buscar la forma más sencilla de implementar cada una de ellas.

2.10.1.2 Panning

Existen tres formas de realizar el *paneo* de una imagen/gráfica:

1. *Por medio de scroll bars*. La imagen/gráfica es explorada a través de un scroll bar horizontal y de un scroll bar vertical.

2. *Por medio de "ticky hand"* - mano pegajosa -. Consiste en la grabación de una parte del dibujo por medio de la operación "*Mouse down, Mouse up*", es decir, la parte del dibujo es seleccionada cuando el botón del mouse es presionado y hasta que el botón es liberado.

3. *Por medio de "flechitas"*. A través de estas teclas el usuario puede desplazarse horizontal y verticalmente a través de la gráfica.

En algunos casos puede restringirse el *paneo*, por ejemplo cuando en la vista detallada existen confusiones entre partes similares de una gráfica, como es el caso de un doctor cuando desea examinar minuciosamente una sola vértebra de la columna vertebral, si en este caso se presentará la opción de *paneo*, el usuario podría sufrir confusiones entre cual vértebra esta actualmente sobre la pantalla.

2.10.2 OPERACIONES AUTOMÁTICAS

Son las operaciones que realiza por si solo el *browser*. Este tipo de operaciones le permite al usuario concentrarse más en las tareas principales: análisis, monitoreo, diagnóstico de la gráfica, etc. Entre estas se encuentran salvar puntos, navegación, manejo de ventanas y bosquejo de la imagen

2.10.2.1 Salvar puntos

Se refiere a la marcación de puntos sobre la imagen sobre las locaciones de interés, lo cual permite retornar rápidamente a estos puntos salvados. Este proceso aumenta grandemente la velocidad de navegación y los diagnósticos.

2.10.2.2 Navegación

Un área puede ser marcada y explorada sistemáticamente. Estas técnicas son útiles para diagnosticar o para simplificar las tareas de exploración.

2.10.2.3 Manejo de ventanas

Se recomienda que cuando el *browser* posea múltiples vistas el sistema coloque las ventanas de forma automática. La coordinación entre éstas puede simplificar enormemente el control sobre su manejo.

2.10.2.4 Bosquejo de la imagen

La identificación automática de las características de la imagen /gráfica facilita el gran cuerpo de trabajo en el despliegue sobre la pantalla de la computadora.

Si el *browser* extrae de forma automática las principales características de la imagen facilita enormemente el estudio de la imagen, es decir, sí extrae y presenta la información de forma automática, las tareas que debe desempeñar el usuario resultarán más sencillas, y posiblemente sea de mayor agrado que tener que realizar manualmente cada una de las operaciones. Por ejemplo, muchas vistas detalladas pueden ser creadas automáticamente o la velocidad de paneo puede ser ajustada de acuerdo a la presencia de las características de interés.

Muchas automatizaciones son posibles y los recursos son necesarios para determinar los beneficios de tales automatizaciones. Por lo general las operaciones automáticas son comúnmente tareas dependientes que funcionan en browsers especializados.

Un estudio cuidadoso de las tareas que debe desempeñar un browser y las herramientas mas simples que lleven a cabo serán de gran utilidad para su implementación. Algunos aspectos que se deben considerar son: los recursos dentro de los cuales se va a desempeñar (velocidad de acceso de la información, velocidad de despliegue, tamaño de la pantalla, etc.), y la realización de experimentos controlados que mejoren la velocidad y radios de error. La elegancia y legibilidad son importantes para el acceso del público, debido a que si la presentación es agradable para el usuario, el *browser* tendrá más aceptación.

El diseño de los browsers de 2D no esta completamente determinado, por lo que es un tópico lleno de vida y el diseñador puede auxiliarse de su experiencia para determinar las características que debe poseer el *browser*.

3. CARACTERÍSTICAS BÁSICAS DE LA INFORMACIÓN PETROLERA²

En este capítulo se describirán a grandes rasgos los antecedentes y las características que posee la información petrolera, principalmente en el área de registros de pozo. Se comienza por citar la definición que R. Debrantes³ hace de un registro de pozo : *“Un registro de pozo es toda aquella obtención gráfica de una característica de las formaciones atravesadas por un pozo en función de la profundidad”*.

3.1 HISTORIA DE LOS REGISTROS DE POZO

La historia de los registros de pozo se remonta al año 1927, cuando fue realizado el primer registro eléctrico en un pequeño campo petrolero de Pechelbronn, Alsacia, provincia del noreste de Francia. Este registro era una única gráfica de resistividad eléctrica de las formaciones rocosas atravesadas por el pozo. Este registro se realizó por el método de “estaciones”, en el cual el instrumento de medición (conocido como sonda) se detenía en intervalos periódicos dentro del agujero, se hacían las mediciones, y la resistividad (valores obtenidos por la sonda), se trazaba manualmente en una gráfica.

A partir de este acontecimiento, la adquisición de registros por medio de sondas para la elaboración de registros de pozo para su posterior interpretación son de gran utilidad dentro de la industria petrolera. Por medio de ellos es posible determinar las características de las formaciones.

3.1.1 Cronología de los avances en registros de pozo

1927	Primer registro eléctrico.
1929	Comercialización del registro de resistividad eléctrica en Venezuela Rusia y E.U.A.
1931	Inclusión del SP (Potencial Espontáneo) con la curva de resistividad en el registro eléctrico. Perfeccionamiento de un método de registro continuo. Desarrollo del primer trazador gráfico.
1936	Introducción de la cámara con película fotográfica. En ese entonces, el registro consistía de la curva de SP y las curvas de resistividad normal corta, normal larga y lateral.
30's	Comienzo del desarrollo del registro de echados con la herramienta de echados anisótropa.
1943	Introducción de la herramienta de echados con tres brazos, junto con un fotociclómetro. Determinaba la dirección y ángulo de inclinación de los estratos.
1946	Reemplazo de sensores de SP por instrumentos de resistividad

	corta. Esto hizo posible la medición del echado en pozos en los que el SP proporcionaba pocos datos correlacionales.
1950	Introducción de la herramienta Microlog para definir capas permeables.
1951	Introducción de la herramienta laterolog, el primer dispositivo enfocado para medir la resistividad profunda.
1953	Desarrollo de la herramienta microlaterolog para lodos salados. El registro de microproximidad y el registro MicroSFL aparecieron mas tarde.
1954	Sustitución de la sonda de cinco bobinas por un arreglo de seis bobinas capaz de realizar mediciones a mayor profundidad
1956	Combinación de un aparato de inducción con cinco bobinas y una curva de SP para formar la herramienta eléctrica de inducción.
1957	Probador de formaciones. Recuperación de una muestra de los líquidos de la formación y medición de la presión del pozo durante el proceso de muestreo.
1962	Introducción de la herramienta de medición de porosidad neutrónica SNP.
1963	Introducción del registro de doble inducción DIL. Realiza las mediciones de inducción profunda, inducción media y resistividad somera.
1970	Introducción de la herramienta de registro neutrónico compensado CNL. El neutrón es aceptado para medir la porosidad.
1978	Herramienta de propagación electromagnética EPT.
1985	Herramienta de propagación profunda DPT.

3.2 OBTENCIÓN DE LOS REGISTROS DE POZO

Los registros geofísicos de pozo son mediciones de propiedades físicas (resistividad, radioactividad, porosidad, velocidad del sonido, etc), realizadas a través de las formaciones de un pozo. Fig. 3.1

La toma de registros se lleva a cabo desde una cabina especial, montada en un camión denominada "laboratorio móvil", la cual debe contener entre otros aditamentos :

- Los instrumentos necesarios de medición de fondo, cable eléctrico y un malacate que se necesita para bajar las herramientas al pozo.
- El equipo de superficie necesario para alimentar las herramientas de fondo.
- El equipo de superficie necesario para recibir, procesar y efectuar una grabación permanente del registro.

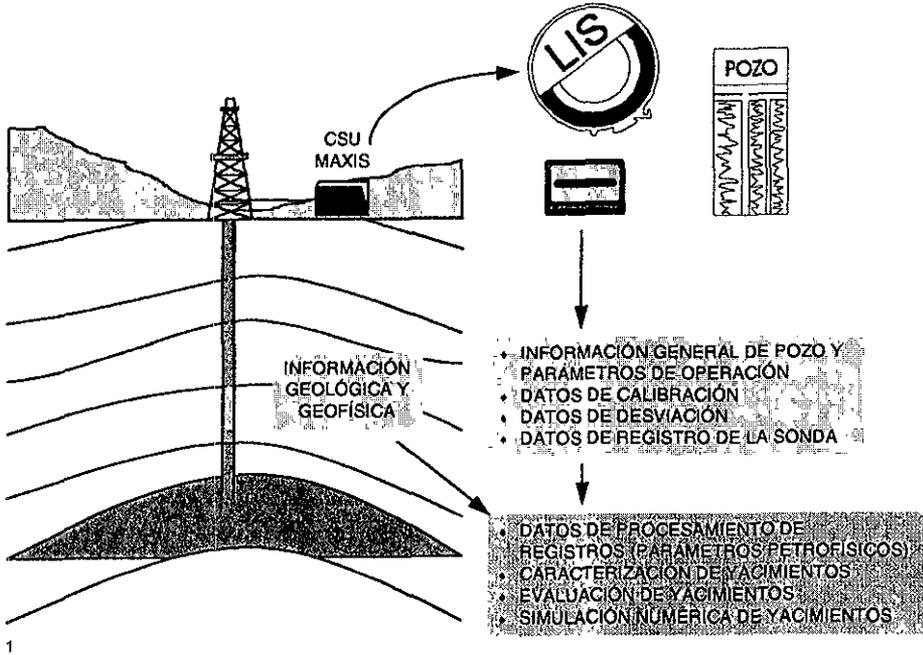


Fig. 3.1. Obtención de los registros de pozo.

Las mediciones se llevan a cabo mediante dos elementos :

1. **Sensores o sonda.** De acuerdo al tipo de medición que se efectúe es empleado cierto tipo de sensor. Por ejemplo : los sensores de resistividad usan electrodos o bobinas, los sensores acústicos emplean transductores sónicos, los sensores de radioactividad emplean detectores sensibles a la radioactividad, etc. La funda de la sonda puede ser de acero o de fibra de vidrio.
2. **Cartucho.** El cartucho contiene los elementos electrónicos que alimentan a los sensores que procesan las señales de medición resultantes y que transmiten las señales por el cable hacia la superficie. El cartucho puede ser un componente independiente que se atornilla a la sonda para formar así la herramienta completa, o bien puede combinarse con los sensores en una sola herramienta. La funda del cartucho es generalmente de acero.

La herramienta o herramientas se conectan a un cable eléctrico cubierto con un armazón de acero que proporciona la fuerza suficiente para soportar el peso de la herramienta y jalarla en caso de que se atore. Los registros se realizan normalmente durante el ascenso del pozo con objeto de asegurar la tensión del cable y llevar un mejor control de la profundidad.

La transmisión de las señales por el cable puede realizarse de forma analógica o principalmente en forma digital.

El equipo de superficie recibe las señales de la herramienta, procesa y/o analiza estas señales. Así las señales deseadas se registran en cinta magnética en forma digital, se pueden observar en tubos catódicos y grabarse en película fotográfica de manera analógica. La película fotográfica se procesa en la unidad y copias de papel impreso se preparan a partir de ella. A la grabación continua de estas señales de adquisición se le denomina **registro**.

El procesamiento de las señales enviadas por la sonda puede realizarse en tres lugares, cuando la herramienta está aun adentro del pozo, en superficie (en la cabina). O bien en un centro de computo. El lugar depende de donde son necesarios los resultados, por ejemplo puede ser mas útil en algunos casos en las inmediaciones del pozo o bien en centros de cómputo donde se realice el estudio más detallado de los resultados hecho por expertos. En la mayoría de los casos las señales son grabadas en dispositivos de almacenamiento masivo de información, como pueden ser cintas magnéticas, cartuchos de 4 mm, 8 mm, etc. De esta manera los datos originales están disponibles para procesamientos posteriores.

3.3 ALMACENAMIENTO DE LOS REGISTROS GEOFÍSICOS DE POZO

Para el almacenamiento de los registros de pozo se empleó la película fotográfica en los primeros equipos de adquisición de datos analógicos y actualmente es grabada en medios magnéticos como cintas de carrete de ½ " o cartuchos de 4 mm, en un formato estándar de intercambio de información denominado LIS (Log Interchange Standard), la información en este formato puede ser procesada posteriormente en centros de computo, para su utilización en diversos estudios.

El uso intensivo de computadoras personales en distintas operaciones de la industria petrolera a finales de la década de los ochenta, generó la necesidad de un formato de intercambio de información en discos flexibles de 3 ½ ", por lo que la "Canadian Well Log Society" propuso el formato LAS (Log Ascii Standard), ampliamente aceptado.

PEMEX dispone actualmente de un volumen importante de registros geofísicos, en cintas de ½", cartuchos de 4mm, y películas fotográficas almacenadas en instalaciones propias y en Centros de Proceso de Registros Geofísicos del Instituto Mexicano del Petróleo.

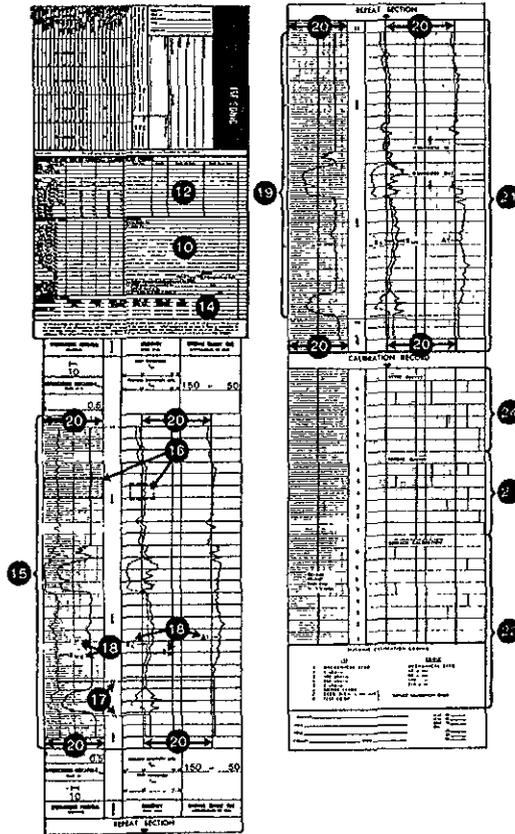


Fig. 3.2 Forma de presentación de una gráfica que muestra un registro.

El flujo de información actual es mostrada a grandes rasgos en la Fig. 3.3, donde la información adquirida se encuentra en formato LIS, el cual permite grabar la operación de toma de registros en una cinta o cartucho, cuya contenido equivale a una gráfica como la mostrada en la Fig. 3.2, el manejo de la información se realiza a través de equipos especializados(estaciones de trabajo), que permiten su visualización, procesamiento y conversión a otros formatos. Debido al alto costo de este tipo de equipos, PEMEX y el IMP no disponen de la cantidad necesaria de ellos, y además en el caso de que los adquiriera, requeriría de fuertes inversiones para la capacitación del personal.

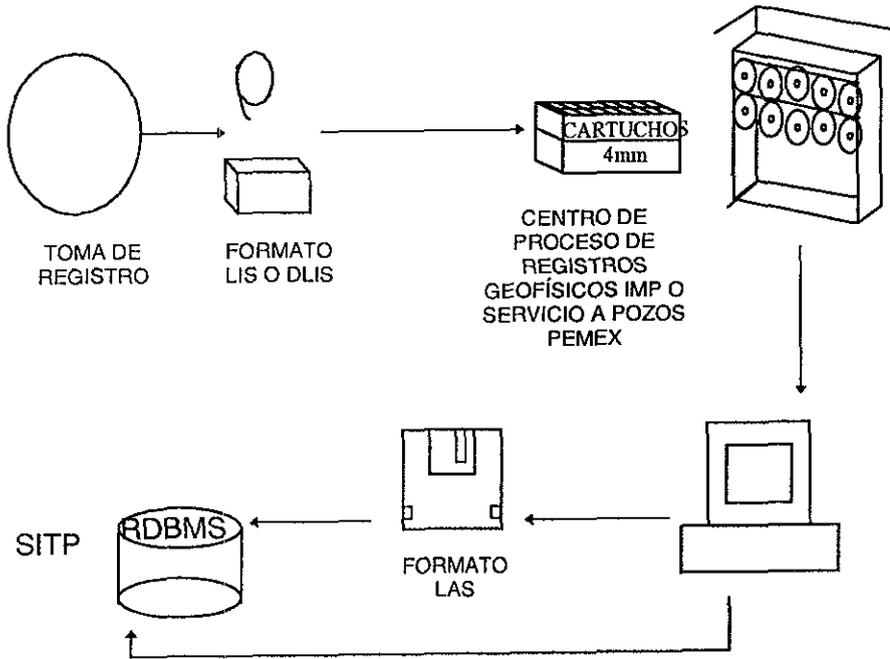


Fig. 3.3. Flujo de información en la obtención de registros de pozo.

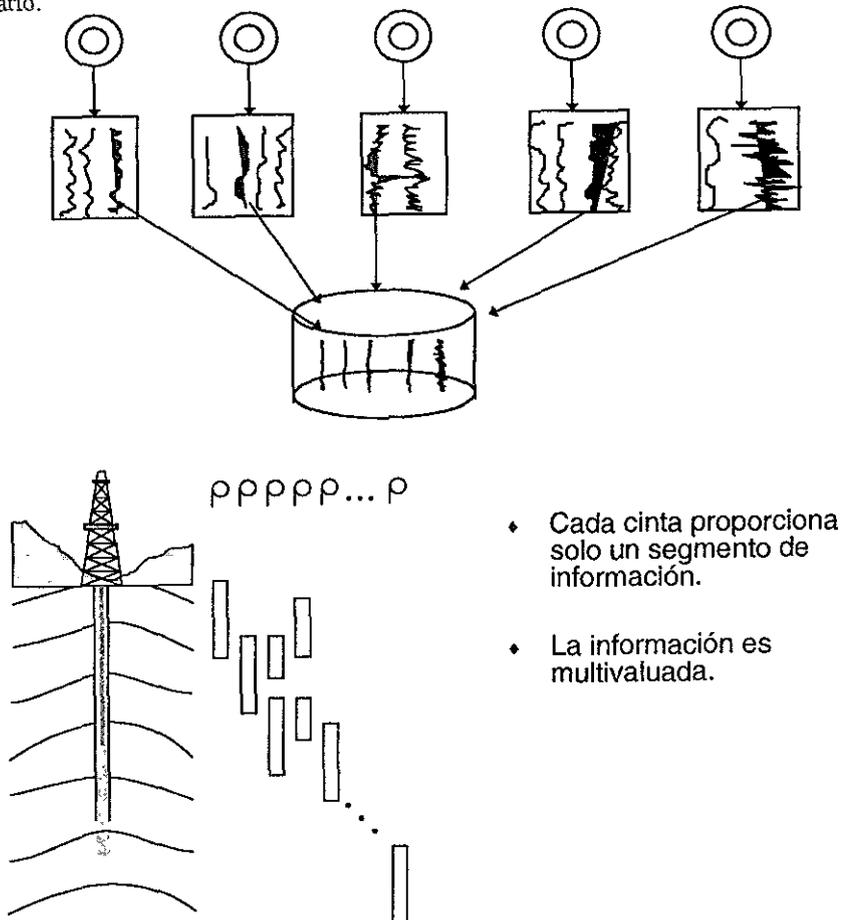
3.4 CONTENIDO DE UN REGISTRO

Un registro presenta los siguientes elementos de información de acuerdo al estándar API del formato de encabezados de registros (Fig. 3.2).

1. Nombre del pozo y compañía.
2. Fecha de operación.
3. Datos de profundidad.
4. Datos del lodo.
5. Temperaturas de agujero.
6. Numero de camión y localización.
7. Tiempo.
8. Calibración.
9. Numero de corrida.
10. Columna de observaciones.
11. Tipo y numero de serie de equipo.
12. Escalas y cambios de escala.
13. Otros servicios.
14. Calibración de datos.

- 15.-21. Presentación de curvas.
- 22. Calibración de superficie.
- 23. Calibración anterior a la operación.
- 24. Calibración posterior a la operación.

La información obtenida en cada operación de toma de registros representa solo un segmento del total del pozo (Fig. 3.4), por lo que es necesario realizar una integración de subsegmentos de interés de cada una de las cintas, cartuchos o películas fotográficas para formar un archivo que representa el subconjunto total de la información requerida por el usuario.



- ♦ Cada cinta proporciona solo un segmento de información.
- ♦ La información es multivaluada.

Fig. 3.4. Integración de todos los registros (subsegmentos) que forman un pozo.

Un registro geofísico representa un conjunto de pares de valores (m_i, n_{ij}) , donde m es el valor de profundidad y n el valor de una propiedad física, los cuales son manejados en forma general por matrices, donde la primer columna es la profundidad y las demás son los valores de las propiedades física. Fig. 3.5.

$$\begin{bmatrix} m_1 & n_{11} & n_{21} & \dots & n_{j1} \\ m_2 & n_{12} & n_{22} & \dots & n_{j2} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ m_i & n_{1i} & n_{2i} & \dots & n_{ji} \end{bmatrix}$$

Fig. 3.5. Representación matncial de un registro geofísico

La columna de profundidad puede ser definida como un conjunto de valores ascendentes o descendentes positivos continuos, con un rango de 0.0 a 9000.0 cuya diferencia $m_i - m_{(i-1)}$ se denomina intervalo, y de acuerdo a este varía el numero de muestras por metro como se muestra en la Fig. 3.6. Debido a que la información en forma tabular o matricial es muy difícil de analizar el manejo de la información de registros geofísicos en la industria petrolera es realizada a través de gráficas que permiten a los usuarios realizar diferentes análisis y evaluaciones dentro de sus actividades.

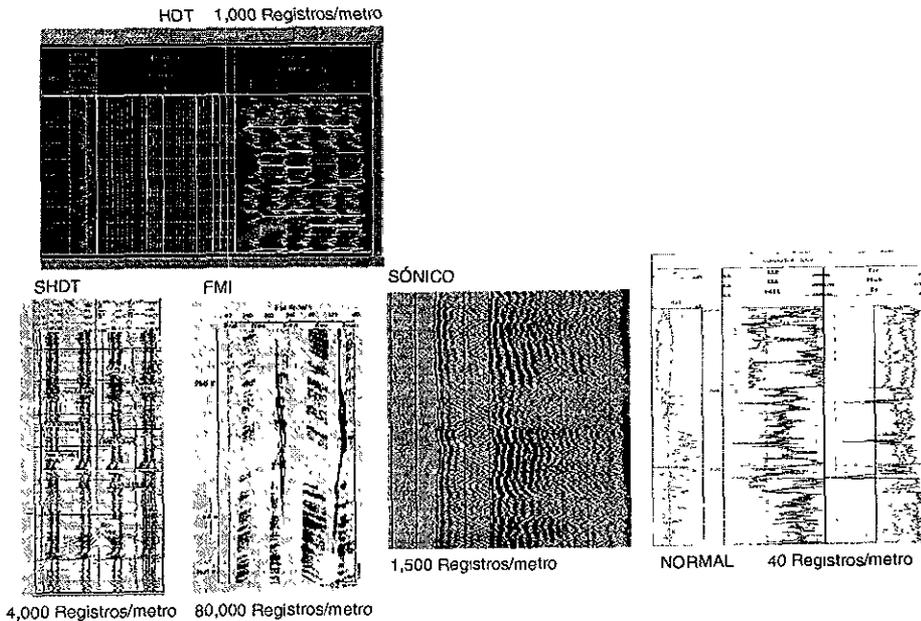


Fig. 3.6. Ejemplos para la representación gráfica de un registro.

Las gráficas de los registros son empleadas por los especialistas en esta área para determinar las regiones de interés para la caracterización de los yacimientos. Por medio del análisis cualitativo y cuantitativo de los datos y las gráficas es posible determinar el nivel de saturación de agua, saturación de aceite, porosidad y muchas otras características de las formaciones geológicas que permiten conocer mejor los yacimientos cortados por el pozo.

Para la graficación de las curvas, el usuario puede indicar cada uno de los diferentes atributos de cada una de las curvas contenidas en el archivo LAS, como pueden ser : color, escala superior, escala inferior, etc. Actualmente estas gráficas se obtienen a través de un sistema denominado TOTALIN o rTotal97, el cual trabaja sobre una plataforma UNIX con estaciones de trabajo HP, pero como en la mayoría de los centros de cómputo del IMP y de PEMEX, no se cuenta con este tipo de equipo y/o sistema operativo, se tiene la necesidad de realizar un sistema para PC que soporte el manejo de este tipo de datos, con todos los elementos que maneja el sistema actual, y además con la inclusión de nuevas opciones de control y manejo de la información.

4. ANÁLISIS Y DISEÑO DEL BROWSER

4.1 ANÁLISIS

El análisis tiene como propósito la identificación del problema para realizar una descripción clara y consistente que permita plantear una solución adecuada del sistema.

Las técnicas de análisis orientadas a objetos permiten modelar el sistema a partir del modelo del mundo real cuando este se encuentra bien definido, si no es así, como es el caso de muchos sistemas en desarrollo de tecnología, el analista debe proponer un modelo en base al conjunto de requerimientos establecidos dentro del dominio para iniciar la fase de modelado del sistema, un factor que ayuda en el desarrollo de tales sistemas es la experiencia del analista en el dominio de la aplicación.

4.1.1 Definición del problema

El Instituto Mexicano del Petróleo (IMP) cuenta con varios tipos de archivos gráficos y/o formatos de datos que representan la información de registros geofísicos de pozo: archivos LAS, PRN, PS, HPGL, etc; por lo que es necesario la implementación de un sistema para realizar su manejo. Consistiendo en la visualización por medio de un dispositivo de salida (monitor de computadora) y la modificación de las características de la gráfica de datos de registros geofísicos de pozo.

4.1.2 Objetivo

Desarrollar un modelo que permita el manejo, la visualización y modificación de una gráfica de datos generada a partir de un archivo LAS dentro de un dispositivo de salida. El modelo debe presentar una solución en el que el parámetro de mayor importancia sean el tiempo de proceso, la presentación de la información y la interactividad usuario-sistema.

4.1.3 Metodología empleada

Los métodos que pueden emplearse para resolver el problema especificado anteriormente son muy variados. Actualmente la mayoría de los sistemas de cómputo tienden a emplear una metodología de análisis y diseño orientada a objetos, por las ventajas que representa desarrollar un sistema bajo esta metodología: módulos y clases reutilizables, mayor productividad, seguridad (debido al ocultamiento de la información), flexibilidad y mantenimiento menos costoso, etc. La metodología que en este caso se emplea se denomina "Desarrollo de Software con Objetos Orientados a Calidad -DSOOC⁸-". Las razones por las que se adoptó este método son:

- Es un estándar que se emplea en el desarrollo de sistemas en la Línea de Negocios, Adquisición y Procesamiento de Información de Pozos del Instituto Mexicano del Petróleo,
- Está derivado de los trabajos de Grady Booch , Ivan Jacobson⁹ y las técnicas de evaluabilidad de Software y los requerimientos impuestos en la norma DoD-2167 del Departamento de Defensa de los Estados Unidos de América. Fig. 4.1 Lo que permite crear software con cierto nivel de confiabilidad y de calidad.
- Provee un procedimiento que realiza todas las fases de desarrollo de un sistema orientado a objetos: análisis, diseño, implementación - programación-, ejecución de pruebas y al mismo tiempo los documentos creados en cada fase se conjuntan para formar el manual del sistema.

La descripción de la metodología DSOOC se presenta a lo largo del presente capítulo, y puede apreciarse de forma general en la fig. 4.2

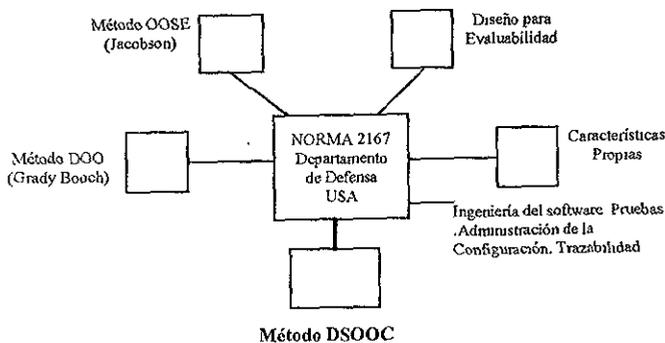


Fig. 4 1.- Contexto del DSOOC.

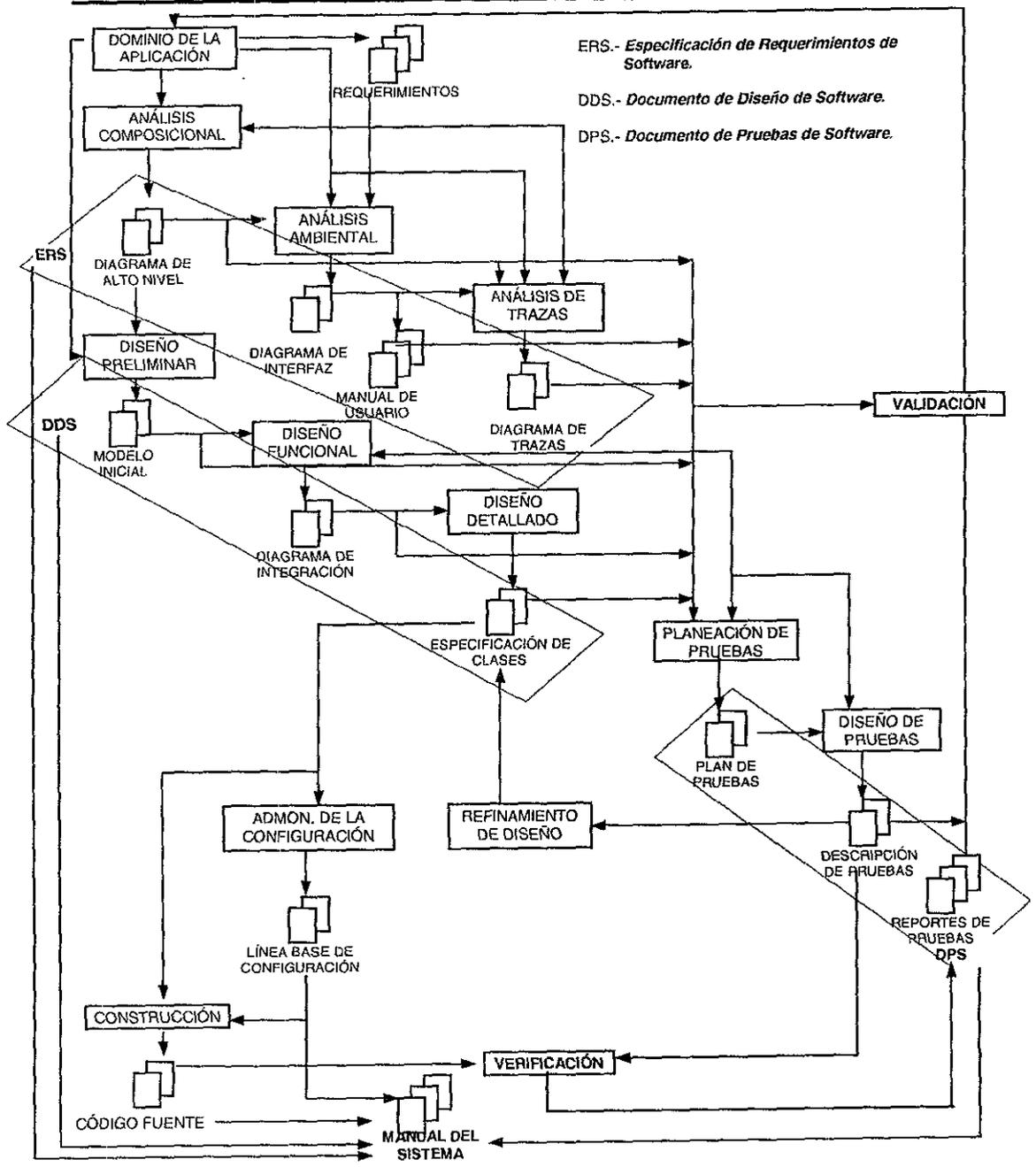


Fig. 4.2. Esquema general de la metodología DSOOC.

4.1.4 Requerimientos del browser para la visualización de la información petrolera (Registros Geofísicos de Pozo)

1. Desplegar archivos de registros de pozos almacenados en formato LAS.
2. El despliegue se hará en computadora personal con sistema operativo windows 95, memoria RAM de 16 MB, monitor SVGA, y espacio disponible en disco duro de al menos 300 MB.
3. Modificación de atributos de las curvas en tiempo de ejecución, de acuerdo a los componentes de la Fig. 4.3:

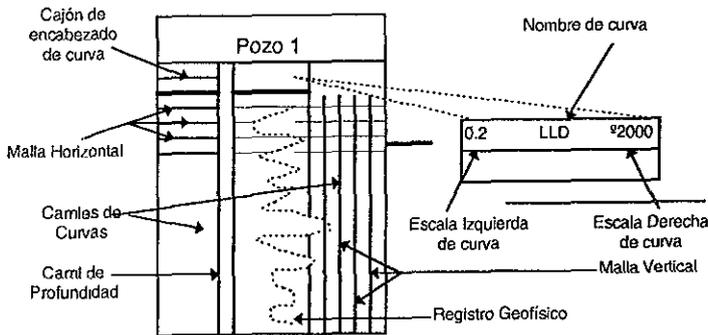


Fig. 4.3. Componentes de la gráfica de un registro geofísico de pozo.

Las modificaciones que se deben realizar a la gráfica del registro son las siguientes:

- **Escala Gráfica del Registro.**- Es la proporción de la profundidad representada en la gráfica. Fig. 4.4

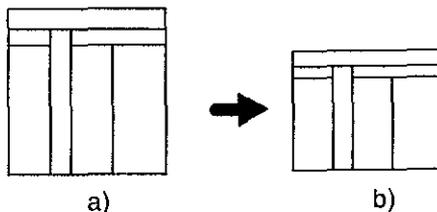
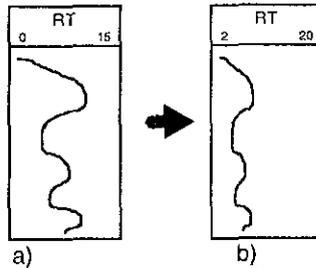
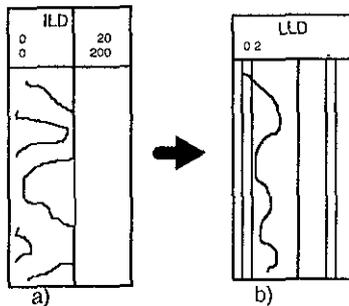


Fig. 4.4. cambio de escala gráfica del registro. a) escala 1:1000, b) escala 1:2000.

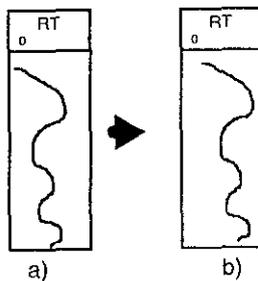
- **Escalas de las curvas.**- Es el rango de valores en el que se encuentra graficada una curva dentro del carril. Se considera una escala izquierda y una escala derecha y en algunos casos se emplean cambios de escala. Fig. 4.5.



- **Tipo de Curva.**- Es la forma de representación o graficación de la curva sea en forma lineal a logarítmica. Fig. 4.6



- **Color.**- Es el color en que se gráfica la curva. Fig. 4.7.



- **Malla de referencia.**- Son líneas guías horizontales y/o verticales que permiten una mejor interpretación de la gráfica. Estas deben ser activadas o desactivadas por usuario. Fig. 4.8.

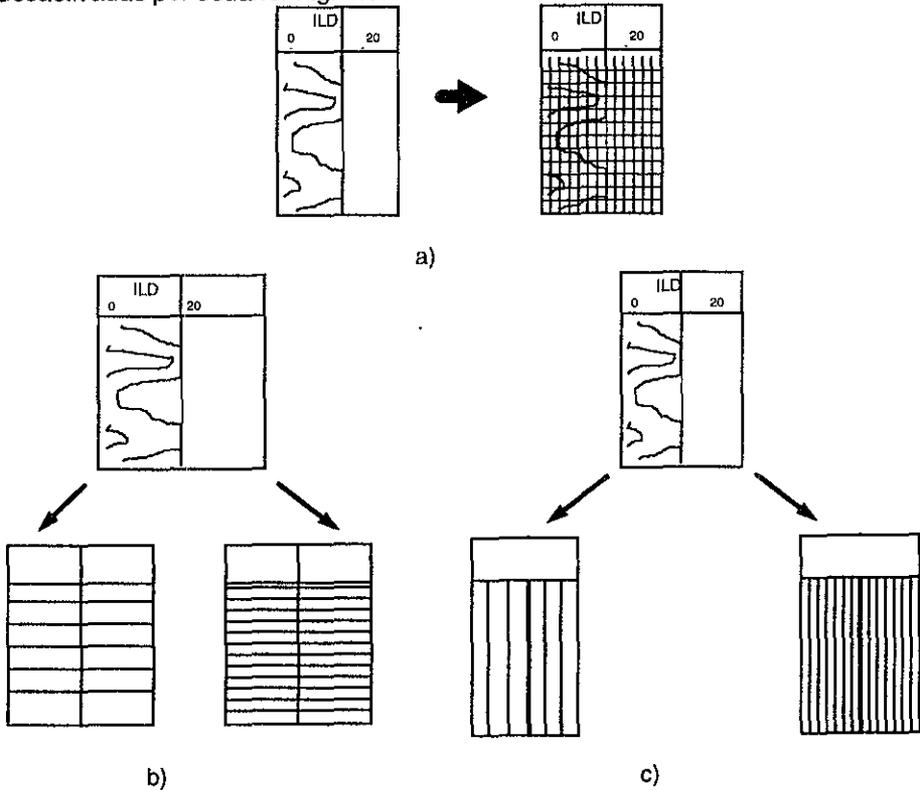


Fig. 4.8. a)Mallas horizontal y vertical, b)Malla Horizontal, c)Malla vertical.

- **Cambio de carril.**- La curva es movida de un carril a otro carril. Fig. 4.9.

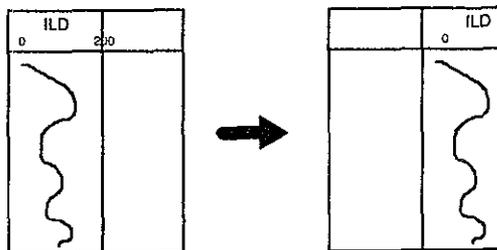


Fig. 4.9. Cambio de carril de una curva.

•**Posición del carril de profundidad.**- El carril de profundidad es movido dentro de la gráfica de una posición a otra. Fig. 4.10

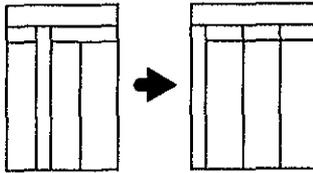


Fig. 4.10 Movimiento del carril de profundidad.

4. **Generación de Metafile.**- Este permite al usuario guardar versiones de trabajo distintas, el almacenamiento se realiza en archivos que contienen todas las características de la gráfica generada.
5. **Zoom out/in.** El sistema debe realizar acercamientos/alejamientos, sujetándose a las necesidades del usuario para poder explorar la gráfica generada en forma más detallada.

4.1.5 ELEMENTOS DE LA GRÁFICA DE DATOS

Una gráfica de datos es la representación de la información de registros geofísicos almacenados dentro de un archivo LAS, cuyo formato se deriva del utilizado en la película fotográfica (FIG. 3.2), la cual sigue un estándar de presentación mostrado en la Fig. 4.1, compuesta por los siguientes elementos :

1. **Cajón de encabezado.**- Utilizado para la identificación del pozo o nombre del archivo.
2. **Cajón de curva.**- Utilizado para la identificación de la curva, los atributos y escalas utilizadas en cada curva, es decir, existe una y solo una curva dentro de un cajón.
3. **Carril de curva.**- Utilizada para la graficación de los datos de la curva. En un carril pueden ser presentadas mas de una curva y por lo general se recomienda que no sean más de tres curvas por carril.
4. **Malla horizontal.**- Se utiliza para la identificación de las marcas de profundidad.
5. **Malla vertical.**- Se utiliza para la identificación de marcas de valor de datos de las curvas, esta puede ser representada lineal o logarítmicamente.

Todos los elementos son controlados por dos conjuntos de datos : Características generales de la gráfica y atributos de las curvas.

4.1.6 CARACTERÍSTICAS GENERALES DE LA GRÁFICA

Las características generales dimensionan la gráfica total y están compuestas por :

- Escala vertical.
- Escala horizontal.
- Número de carriles.
- Número cajones.
- Intervalo de graficación.
- Número de curvas a graficar.

El ancho de la gráfica (eje "x") es obtenido en función de la escala horizontal y el número de carriles, los cuales dependen del número de curvas a graficar ; considerando que una curva puede ocupar uno o dos carriles , y que no es conveniente graficar más de tres curvas en un carril la Fig. 4.11, muestra diferentes anchos de gráfica.

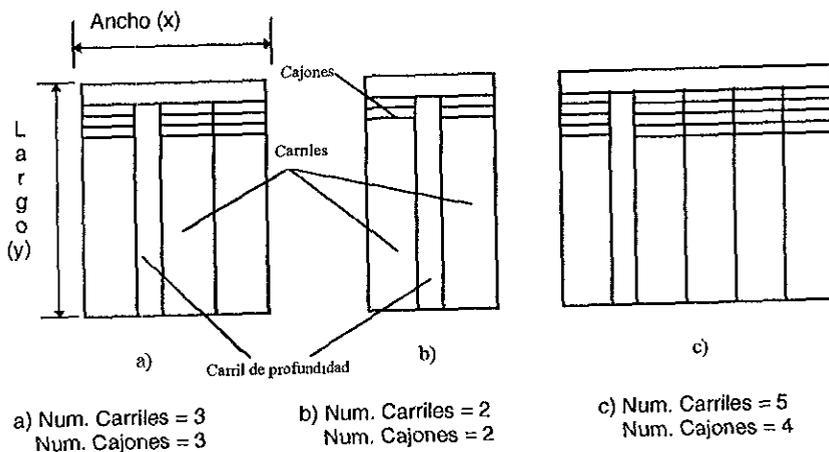


Fig. 4.11. Diferentes anchos de gráfica

El ancho de los carriles utilizando una escala vertical de 1 : 500 es
 ancho de carril de curva : $2 \frac{1}{2}'' = 6.4\text{cm}$
 ancho de carril de profundidad : $\frac{3}{4}'' = 1.8\text{cm}$

La gráfica es dimensionada a lo largo por el intervalo de graficación (el cual va desde 0 a 8,000 m) y la escala vertical, la Fig. 4.12 muestra una tabla de diferentes dimensiones aproximadas en mts. del largo de la gráfica ("y") a diferentes escalas verticales.

INTERVALO	ESCALA VERTICAL					
	1 : 100	1 : 200	1 : 500	1 : 1000	1 : 2000	1 : 5000
0 - 1000	10	5	2	1	0.5	0.2
0 - 2000	20	10	4	2	1	0.4
0 - 3000	30	15	6	3	1.5	0.6
0 - 4000	40	20	8	4	2	0.8
0 - 5000	50	25	10	5	2.5	1
0 - 6000	60	30	12	6	3	1.2
0 - 7000	70	35	14	7	3.5	1.4
0 - 8000	80	40	16	8	4	1.6

Fig. 4.12 Dimensiones de la gráfica según el intervalo de profundidades y la escala vertical.

4.1.7 ATRIBUTOS DE LAS CURVAS

Los atributos de las curvas son un conjunto de valores que determinan la presentación y el dimensionamiento de los datos de registros geofísicos dentro de los carriles de la gráfica, los atributos de cada curva están compuestos por :

- Nombre de la curva.
- Escala Derecha.
- Escala Izquierda.
- Tipo de curva.
- Color.
- Tipo de línea.
- Número de carriles de graficación.
- Cajón.
- Muestreo.
- Malla vertical.
- Malla horizontal.

Estos atributos son únicos a un tipo de registro geofísico (curva) y controlan su presentación en la gráfica.

4.1.8 RANGOS DE CARGA DE OPERACIÓN ESTIMADOS DEL SISTEMA

Las escalas utilizadas normalmente son 1:500 y 1:1000 con intervalos promedio de 4000 mts., un número de curvas igual a 12 distribuidas en 6 carriles, por lo cual se generan graficas de datos con los siguientes rangos de largo y ancho :

- ancho (x) de 0.195 a 0.381 mts.
- largo (y) de 4 a 8 mts.

El número de vectores generados por gráfica es calculado en base al intervalo de muestreo, el cual varía de 0.15 a 0.3048 mts. En un intervalo promedio de 4000 mts son :

- Número de vectores por curva : de 13123 a 26666
- Número total de vectores (12 curvas) : 157 476 a 319 992

4.1.9 ANÁLISIS COMPOSICIONAL

El objetivo del análisis composicional es el obtener un modelo inicial del sistema que sirva como base del diseño final en función de los requerimientos solicitados, el cual normalmente es extraído del dominio de la aplicación, en este caso el modelo real no existe por lo cual debe ser propuesto en función de los requerimientos y la naturaleza del problema a resolver.

4.1.10 CONCLUSIONES DE CARGA DE OPERACIÓN DEL SISTEMA

El punto 4.1.4 reúne el conjunto de requerimientos de carga de operación del sistema al cual debe ser sometido, de los que se derivan las siguientes conclusiones :

A) El número de vectores para una gráfica es elevado y distribuidos en una superficie mayor a la de un monitor de vídeo, por lo que solo es posible visualizar una parte a la vez, lo que conduce a determinar el número de pantallas que se requerirían para visualizar una gráfica de datos promedio de 8.00X0.381 mts. en un monitor de vídeo de 14" (11 pulgadas horizontales y 8 pulgadas verticales), como se muestra en la Fig. 4.13

Por lo que el sistema requiere de un manejador de gran volumen de información que permita la rápida presentación en el monitor de vídeo.

B) La información de los elementos que componen a la gráfica son modificados en tiempo real, por lo que el sistema debe ser diseñado con la flexibilidad requerida para este tipo de operaciones.

C) El usuario requiere un estilo de interacción con el sistema definido como **modifica lo que ve**.

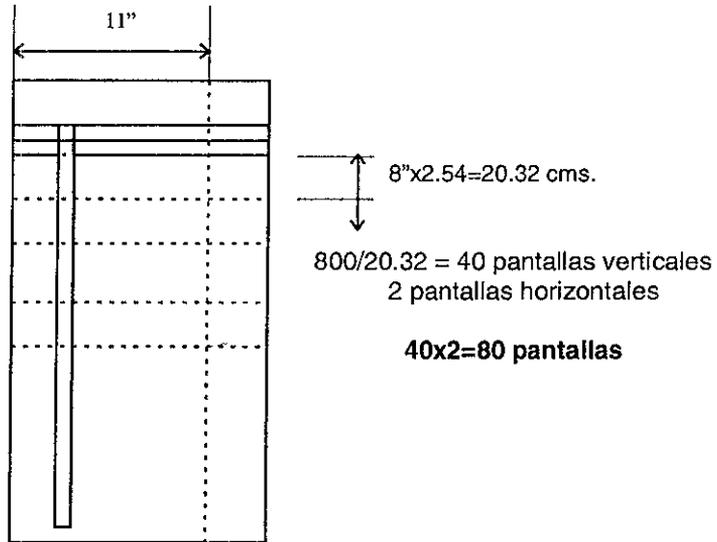


Fig. 4.13. Espacio empleado en una gráfica.

4.1.11 MODELO INICIAL DEL SISTEMA

El modelo inicial propuesto como base del análisis es mostrado en la Fig. 4.14.

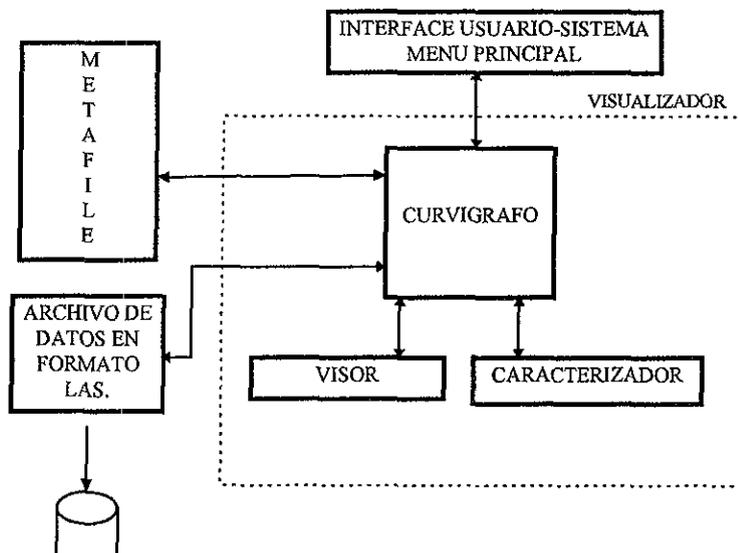


Fig. 4.14 . DIAGRAMA DE ALTO NIVEL.

4.1.12 ESPECIFICACIÓN PRELIMINAR DE COMPONENTES

4.1.12.1 METAFILE

Metafile es un componente que define una gráfica generada a partir de un archivo de datos (en formato LAS), conteniendo todas las características y atributos de las curvas definidos por el usuario. Una vez creado un metafile el sistema debe presentar la gráfica por medio del curvígrafo y del visor.

4.1.12.2 MENÚ PRINCIPAL

Este componente funge como interfaz principal, desarrollado a base de menús , con las principales funciones del sistema.

4.1.12.3 VISUALIZADOR

Es el conjunto de componentes encargado de administrar y controlar los eventos realizados por el usuario para adecuar la gráfica a sus necesidades, además de ofrecer los servicios disponibles por el sistema, el cual es dividido en tres subcomponentes para mejor desempeño :

- CURVÍGRAFO.- Administra y controla los principales eventos relacionados con la información general de la gráfica y de los atributos de las curvas.
- CARACTERIZADOR.- Presenta la información con las características generales de la gráfica de datos.
- VISOR.- Se utiliza como ventana gráfica de despliegue de la gráfica de datos.

4.1.12.4 ARCHIVO DE DATOS EN FORMATO LAS

Es un archivo almacenado con un formato que proporciona toda la información relacionada a un registro de pozo: Propiedades generales (Nombre del pozo, Compañía, Profundidad inicial, profundidad final, nombre de las curvas, etc.) y los datos - lecturas - de cada una de las curvas. Fig. 4.15.

Esta fase conjunta todas las pantallas de interacción del usuario y el sistema en un catálogo que establece las bases para definir las características principales de operación y al mismo tiempo iniciar el manual respectivo (manual de usuario u operación). Para el desarrollo del visualizador primero se desarrollan los elementos de interfaz (EI), los cuales definen un diálogo unitario de interacción con el usuario, de los que se derivan los principales escenarios posteriormente (interfaces).

El objetivo de definir los elementos de interfaz, es el de definir unidades fáciles de manejar, controlar y modificar durante la fase diseño, que además sirvan como base para el desarrollo de un prototipo rápido del sistema. La Fig. 4.16 proporciona una lista de los principales elementos de interfaz utilizados y la Fig. 4.17 una lista de las principales interfaces del sistema con el usuario (escenarios).

ID	NOMBRE	NEMÓNICO
EI1	Menú principal	EI1
EI2	Ventana Principal	EI2
EI3	Selección de Archivo LAS	EI3
EI3.1	Selección de Metafile	EI3.1
EI3.2	Avance de Gráfica	EI3.2
EI4	Guardar archivo	EI4
EI4.1	Guardar archivo como	EI4.1
EI4.2	Petición de sobrescritura	EI4.2
EI5	Confirmación de salida del sistema	EI5
EI6	Barra estándar	EI6
EI7	Barra de Herramientas	EI7
EI8	Propiedades Generales	EI8
EI9	Cajones de la gráfica	EI9
EI10	Modificar propiedades generales	EI10
EI11	Modificar atributos de la Curva	EI11
EI12	Datos del sistema	EI12
EI13	Índice de ayuda	EI13
EI13.1	Contenido de ayuda	EI13.1
EI14	Presentación de la Gráfica	EI14

Fig. 4.16. Tabla de Elementos de Interfaz.

ID	NOMBRE	NEMÓNICO
DI1	Abrir archivo de datos(LAS)	DI1
DI2	Abrir Metafile	DI2
DI3	Cerrar archivo	DI3
DI4	Guardar archivo	DI4
DI5	Guardar como	DI5
DI6	Salir	DI6
DI7	Ver barra de herramientas	DI7
DI8	Ver barra estándar	DI8
DI9	Ver características grales.	DI9
DI10	Ver Cajones	DI10
DT11	Ver Acercamiento/Alejamiento	DT11
DI12	Modificar características grales.	DI12
DI13	Modificar atributos de las curvas	DI13
DI14	Ayuda acerca de	DI14
DI15	Ayuda contenido	DI15
DI16	Ayuda índice	DI16

Fig. 4.17. Tabla de Diagramas de Interfaz.

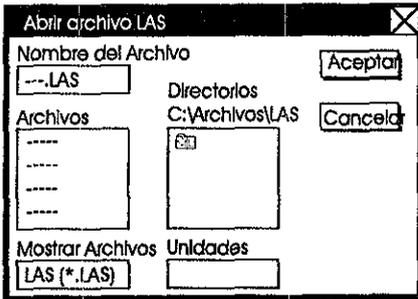
Elementos de Interfaz

VIPREG			
Archivo	Ver	Modificar	Ayuda
Abrir	Barra standar	Prop. grales.	Acerca de
Abrir Metalite	Barra de herramientas	Atrib. de las curvas	Contenido
Cerrar	Características grales.		Índice
Guardar	Cajones		
Guardar como			
Salir			

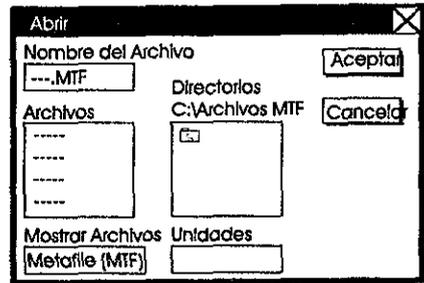
E11 Presentación del Menú principal.

E12 Ventana Principal.

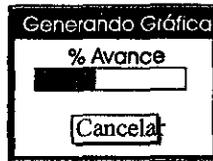
Elementos de Interfaz



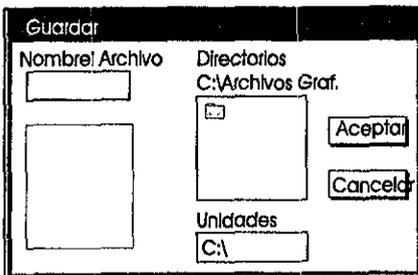
EI3 Selección de un archivo LAS.



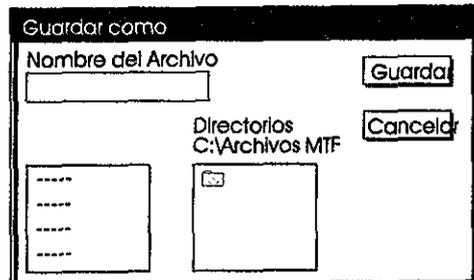
EI3.1 Selección de un Metafile.



EI3.2 Presentación del avance procesado de la gráfica.

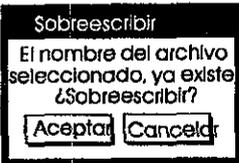


EI3 Guardar un archivo (Metafile).

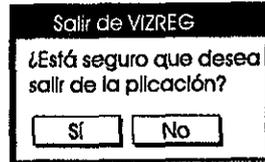


EI4.1 Guardar un archivo (Metafile *.mtf) con un nuevo nombre.

Elementos de Interfaz



E14.2 Sobreescritura del archivo. En caso de que existan uno con el mismo nombre.



E15 Salir de la aplicación.



E16 Barra estándar.



E17 Barra de herramientas.

curva1	p. r o f.	curva7	curva3	
		curva6		curva5
curva2		curva8	curva4	

E18 Cajones de la gráfica.

Elementos de Interfaz

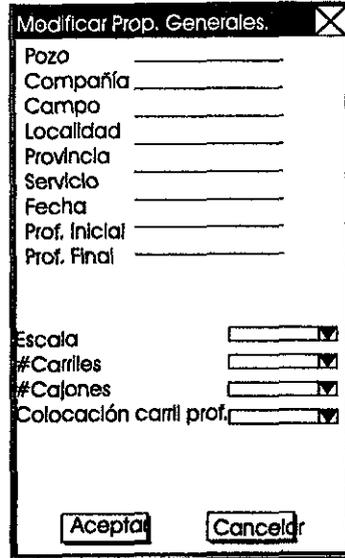


Prop. Generales.

Pozo _____
Compañía _____
Campo _____
Localidad _____
Provincia _____
Servicio _____
Fecha _____
Prof. Inicial _____
Prof. Final _____

Escala _____
#Carriles _____
#Cajones _____
Coloc. carril prof. _____

Nombre	Unidad
_____	_____
_____	_____
_____	_____
_____	_____



Modificar Prop. Generales.

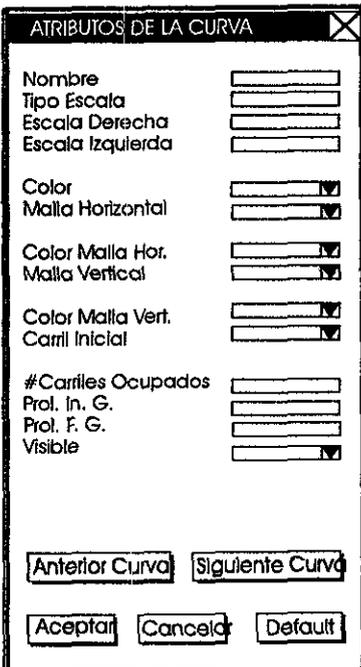
Pozo _____
Compañía _____
Campo _____
Localidad _____
Provincia _____
Servicio _____
Fecha _____
Prof. Inicial _____
Prof. Final _____

Escala ▼
#Carriles ▼
#Cajones ▼
Colocación carril prof. ▼

Aceptar Cancelar

E19 Características Generales (Caracterizador).

E10 Modificar las propiedades generales de la grafica.



ATRIBUTOS DE LA CURVA

Nombre
Tipo Escala
Escala Derecha
Escala Izquierda

Color ▼
Malla Horizontal ▼

Color Malla Hor. ▼
Malla Vertical ▼

Color Malla Vert. ▼
Carril Inicial ▼

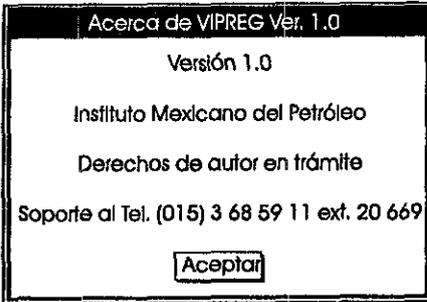
#Carriles Ocupados
Prof. in. G.
Prof. F. G.
Visible ▼

Anterior Curva Siguiete Curva

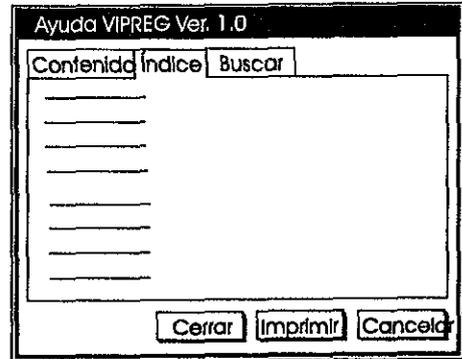
Aceptar Cancelar Default

E11 Modificación de los atributos de cada una de las curvas.

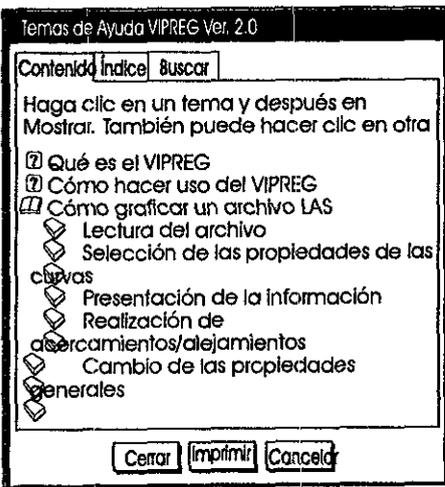
Elementos de Interfaz



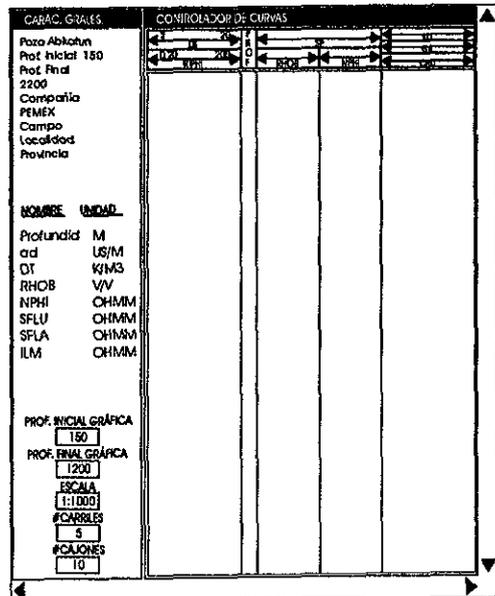
E112 Datos del sistema.



E113 Índice de la Ayuda.

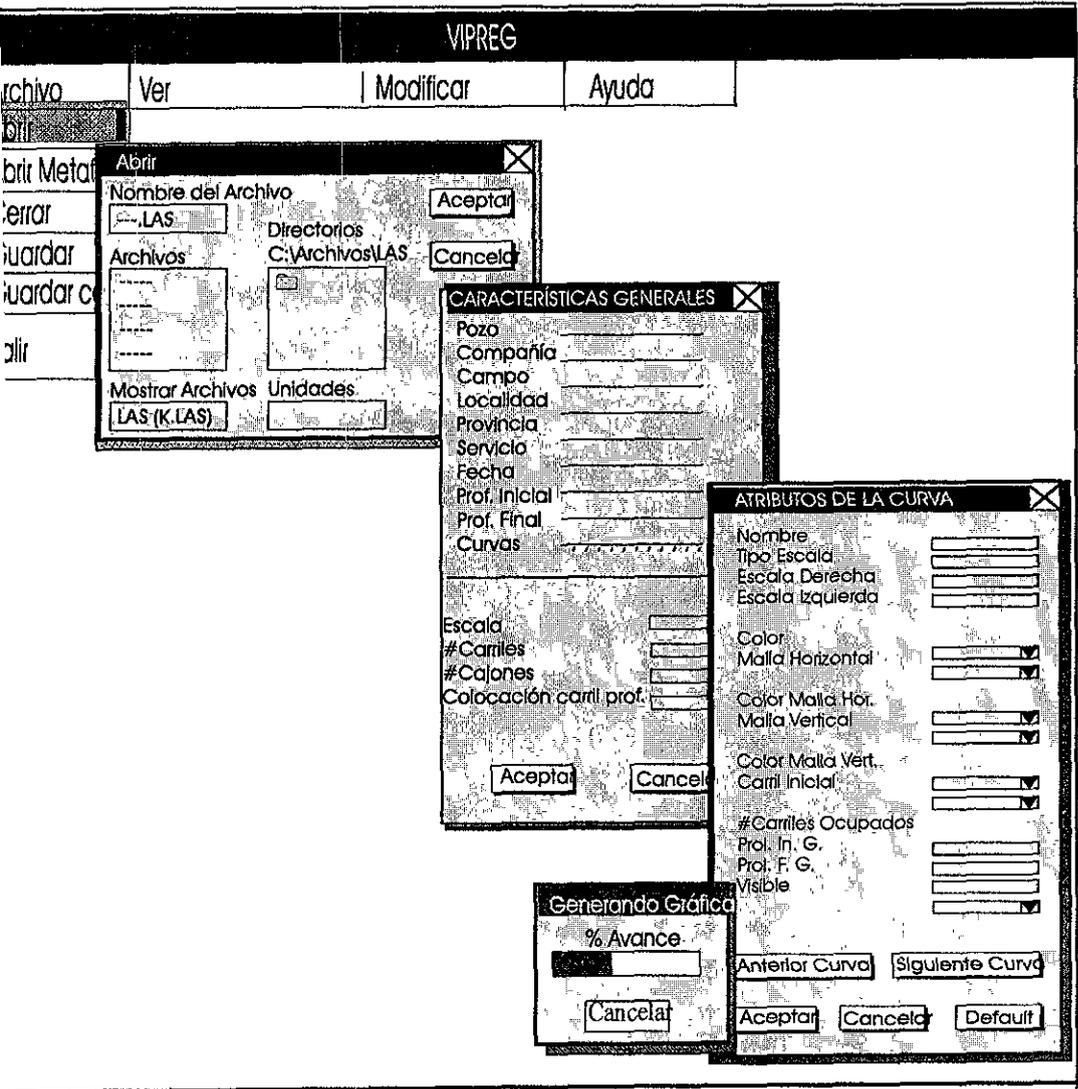


E113.1 Contenido de la Ayuda del sistema.

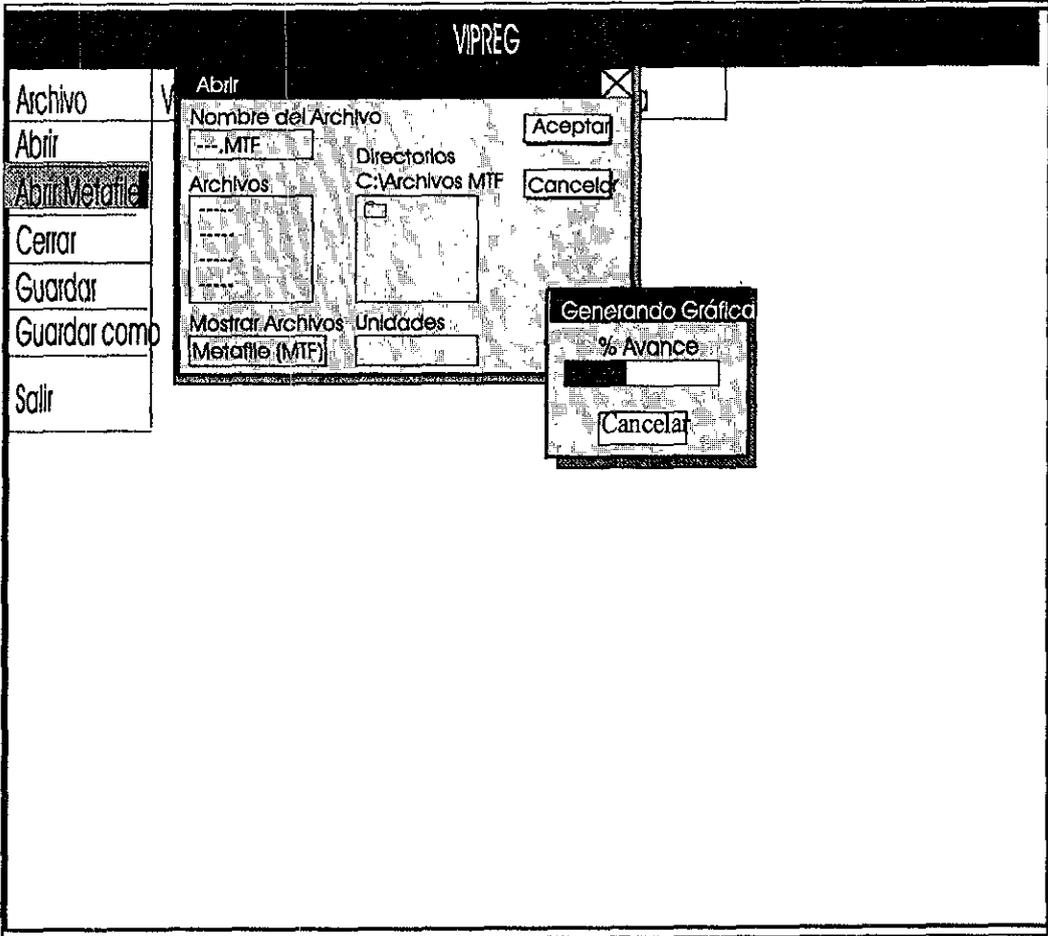


E114 Presentación de la gráfica.

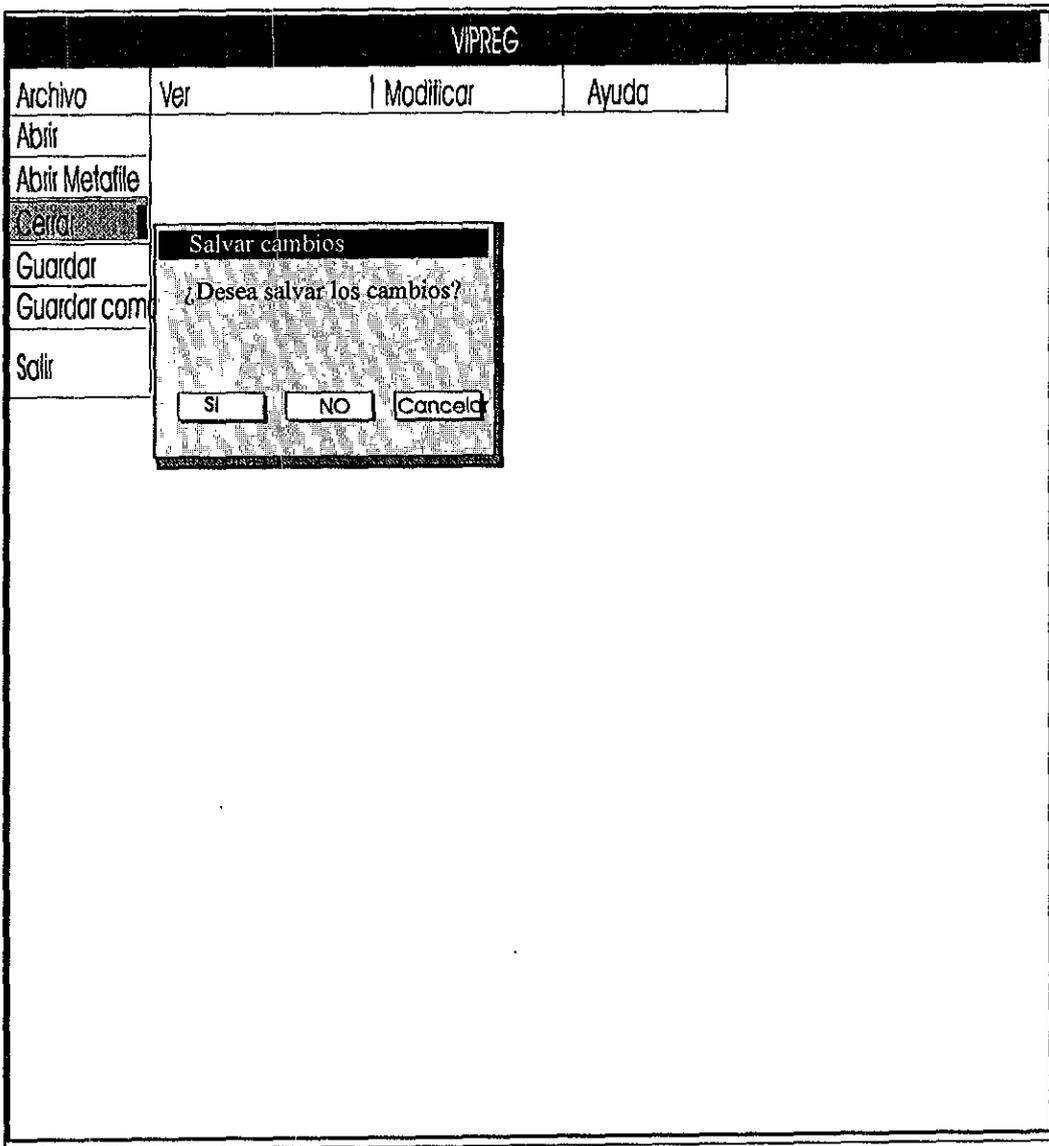
Diagramas de Interfaz



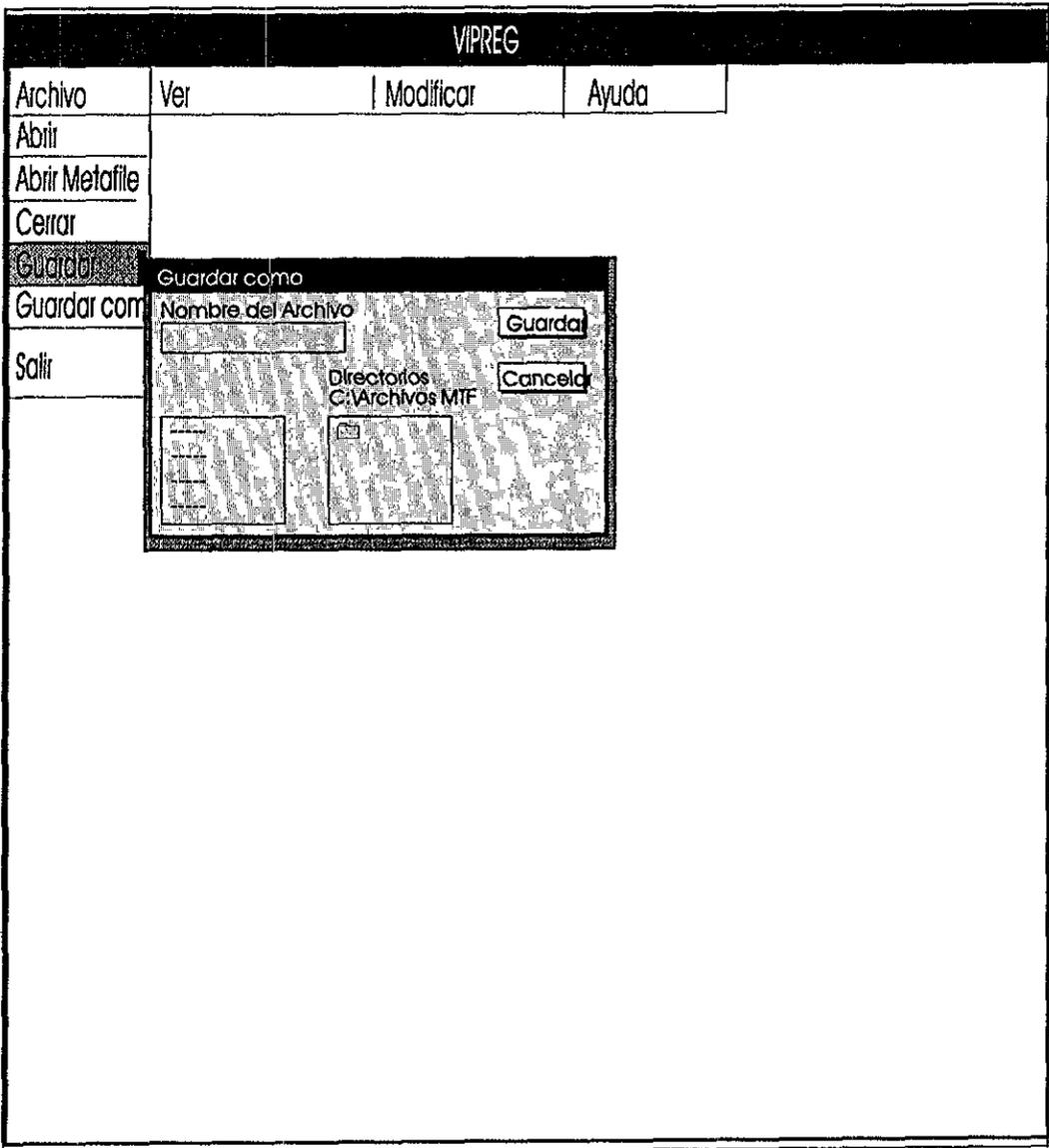
 Abrir Archivo de Datos.



Di2 Abrir Metafile.

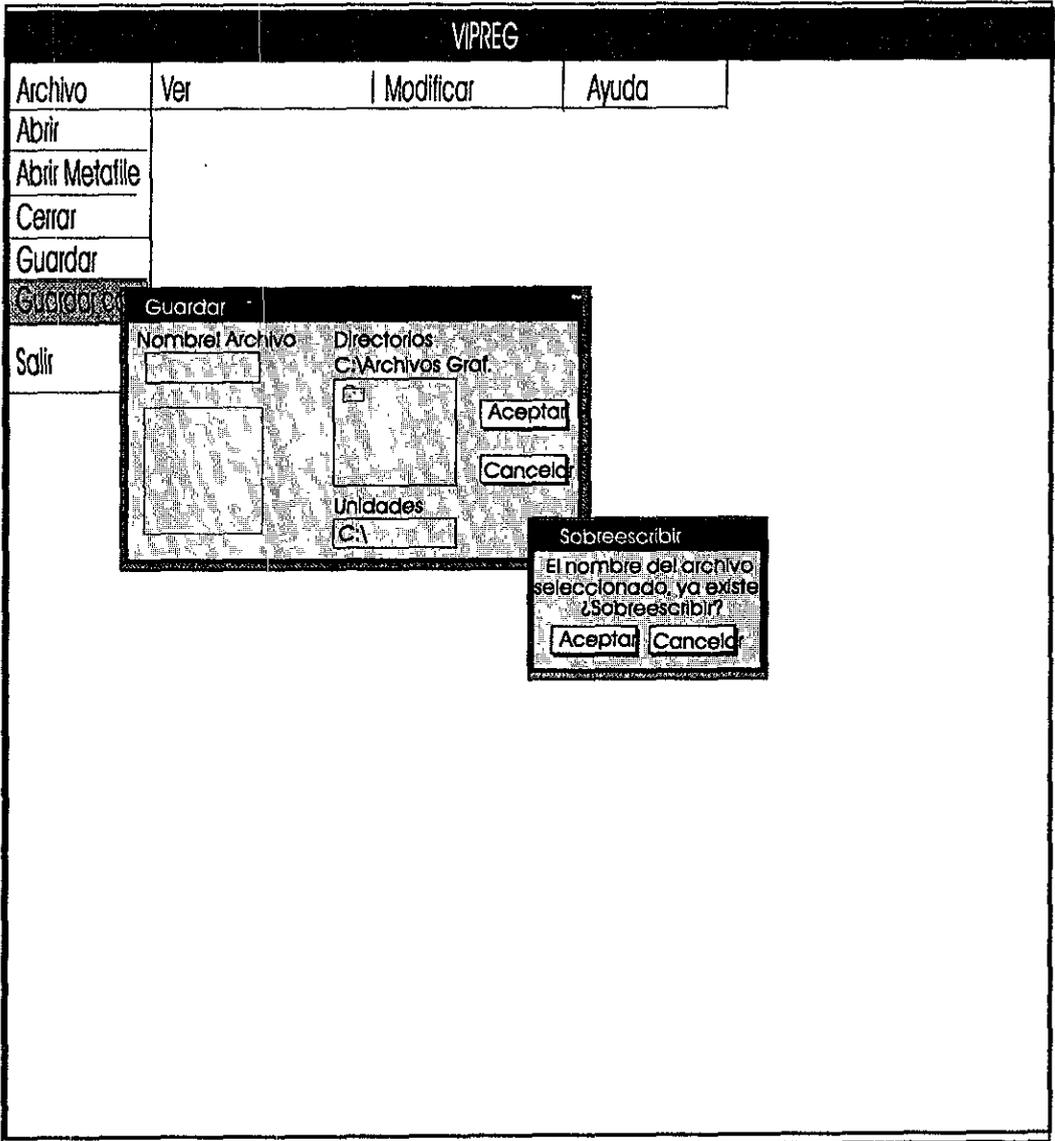


D13 Cerrar Archivo.

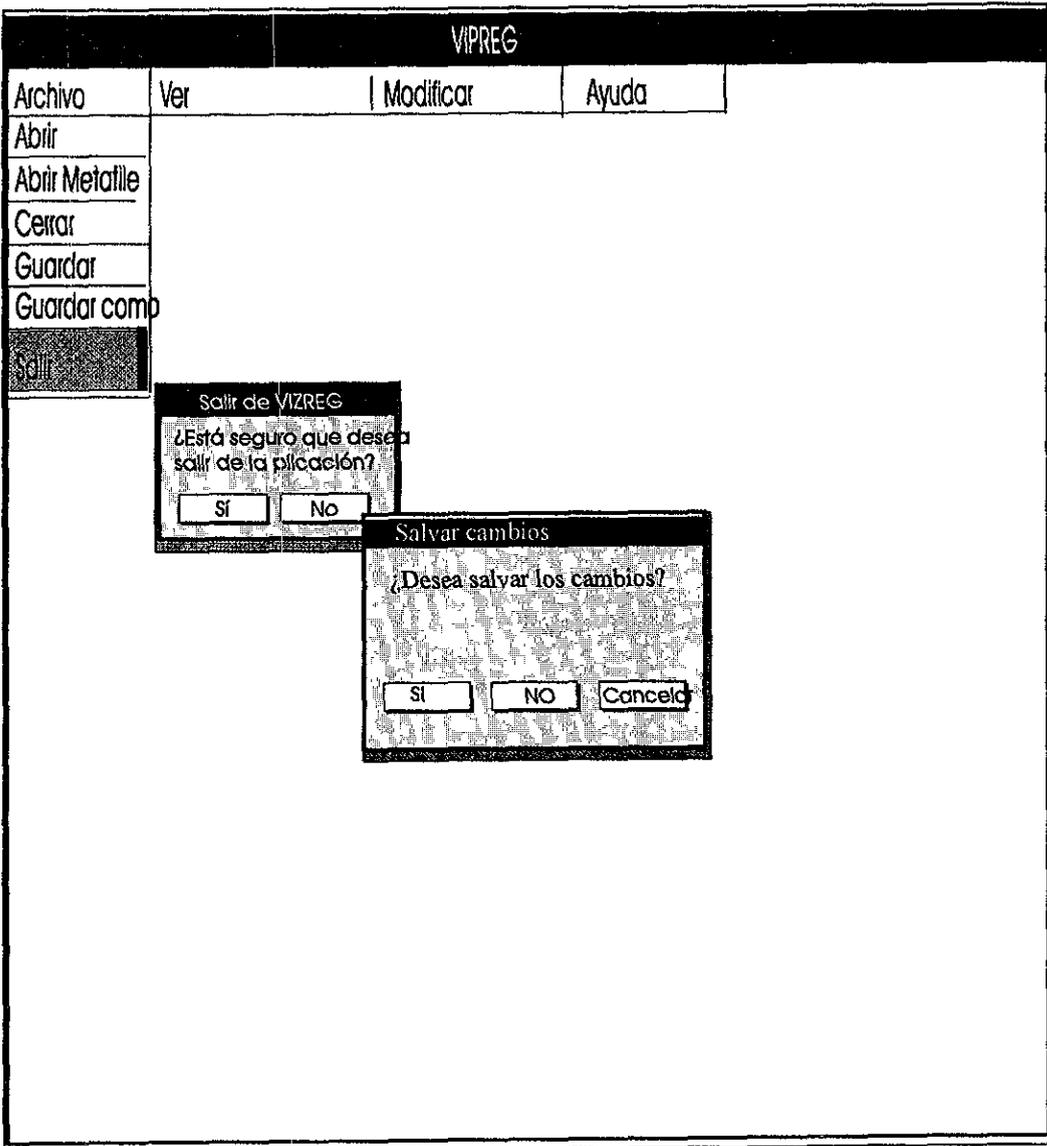


D14 Guardar Archivo.

Diagramas de Interfaz

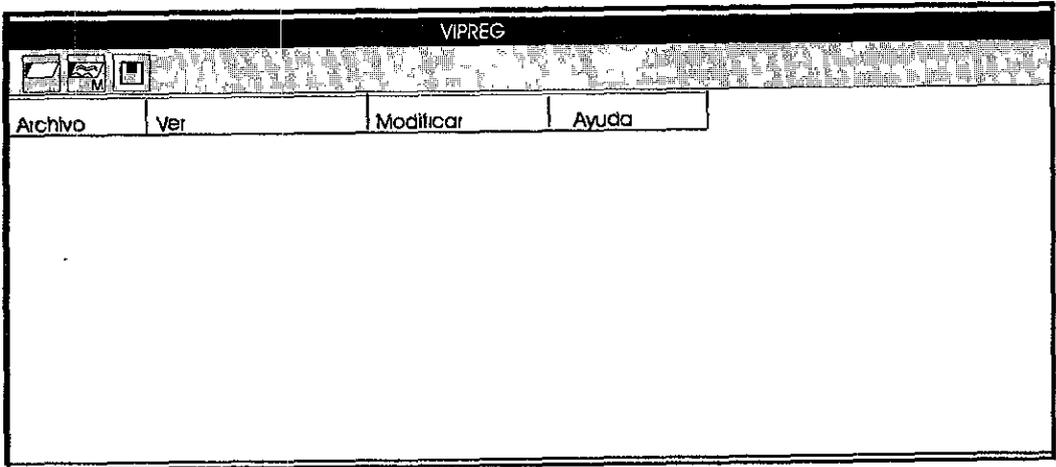
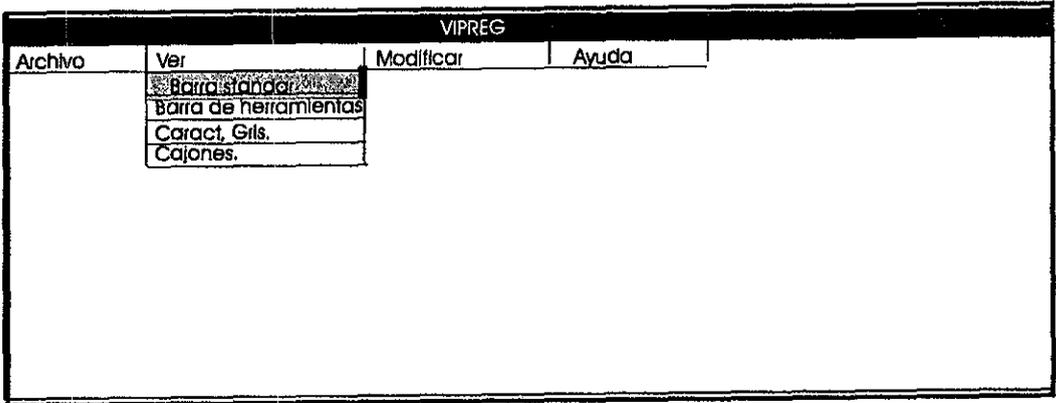


Guardar Archivo Como...



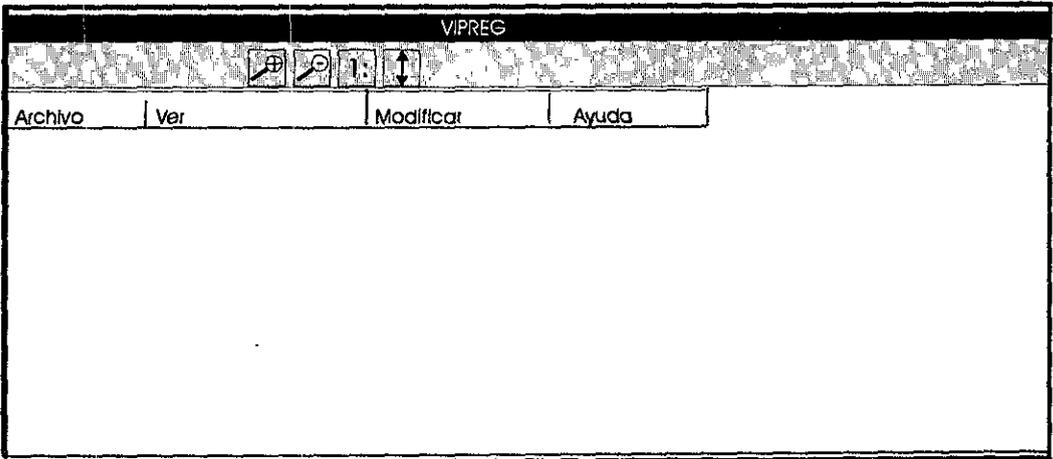
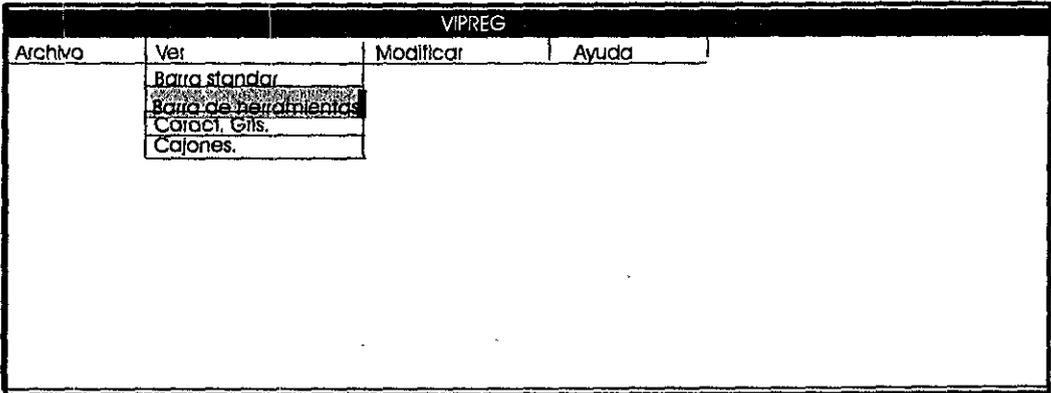
D16 Salir de la aplicación.

Diagramas de Interfaz



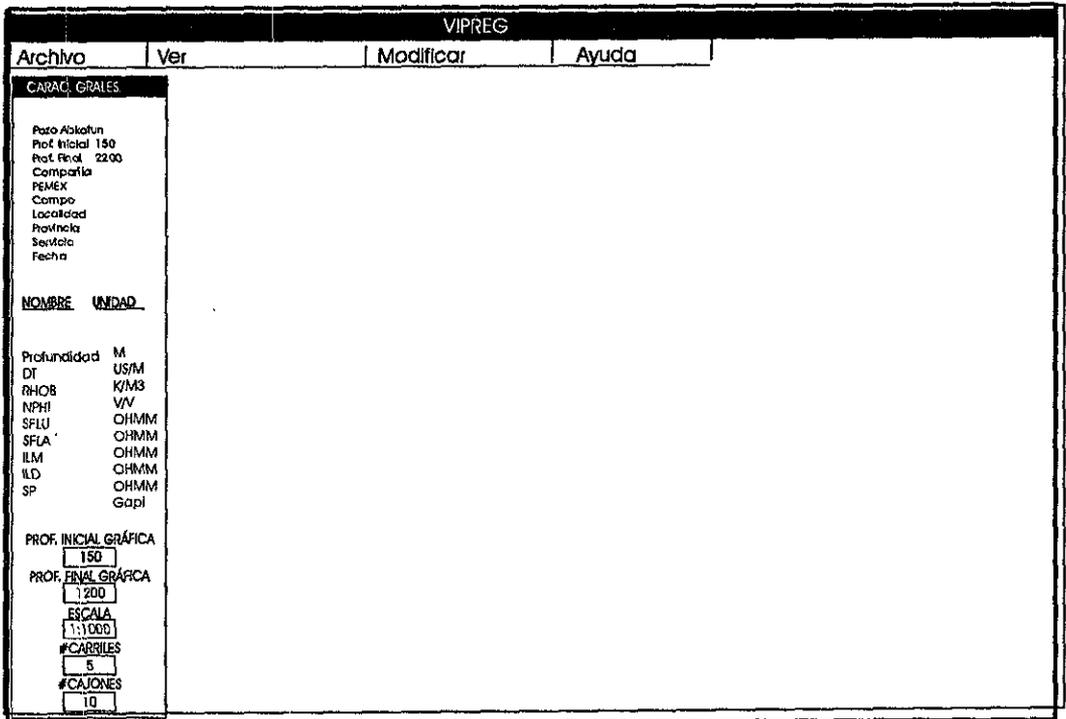
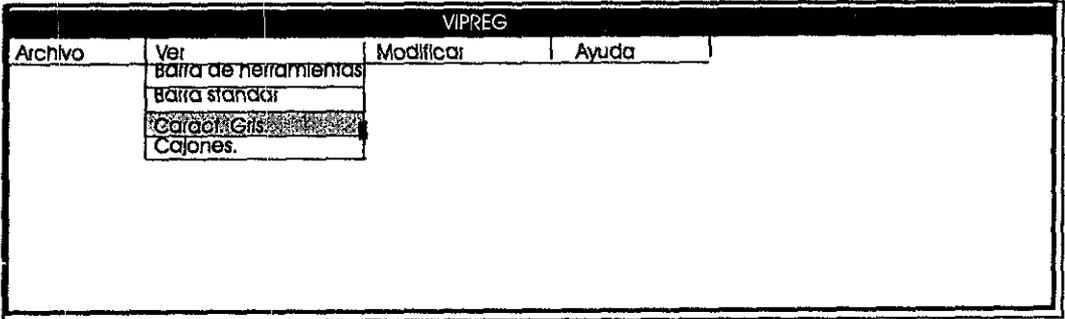
Ver Barra Estándar.

Diagramas de Interfaz



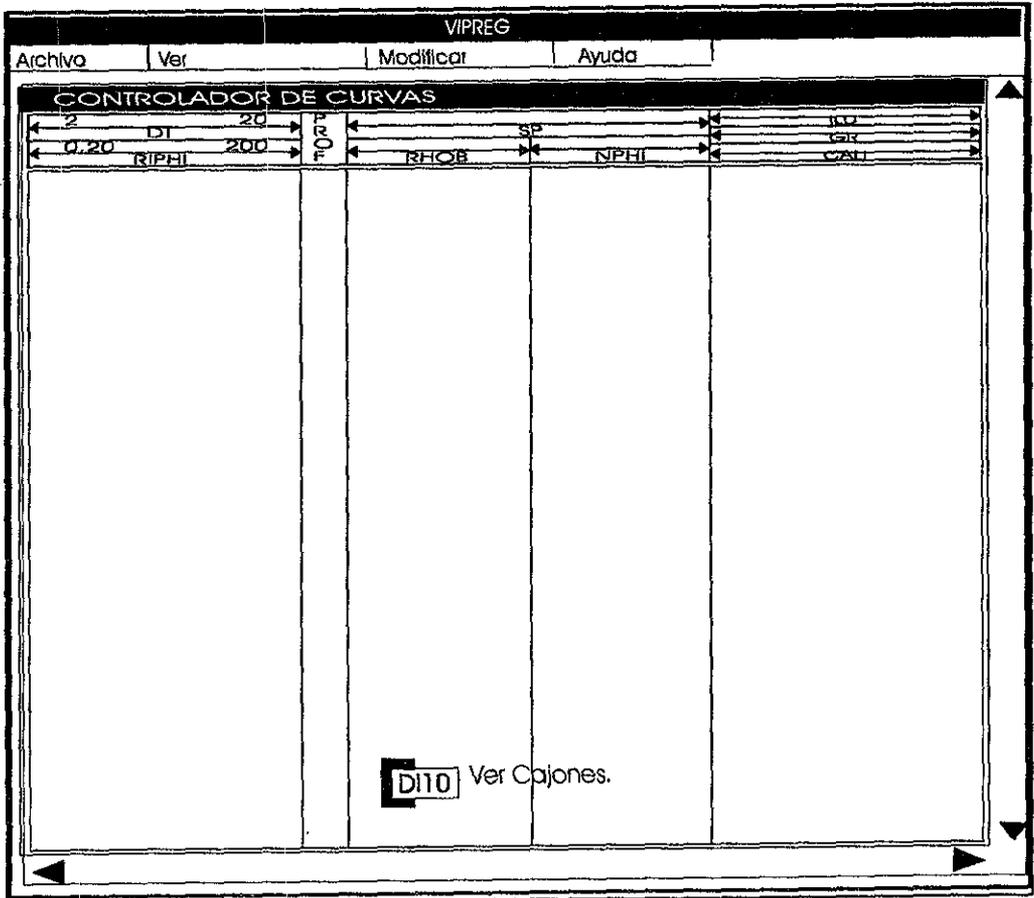
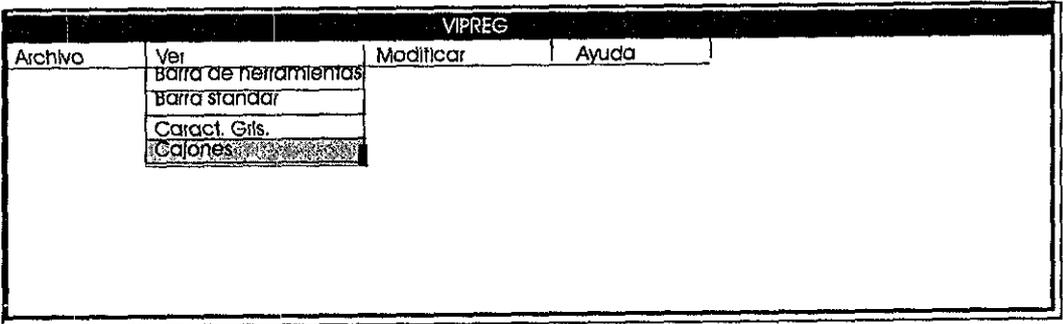
Ver Barra de Herramientas.

Diagramas de Interfaz

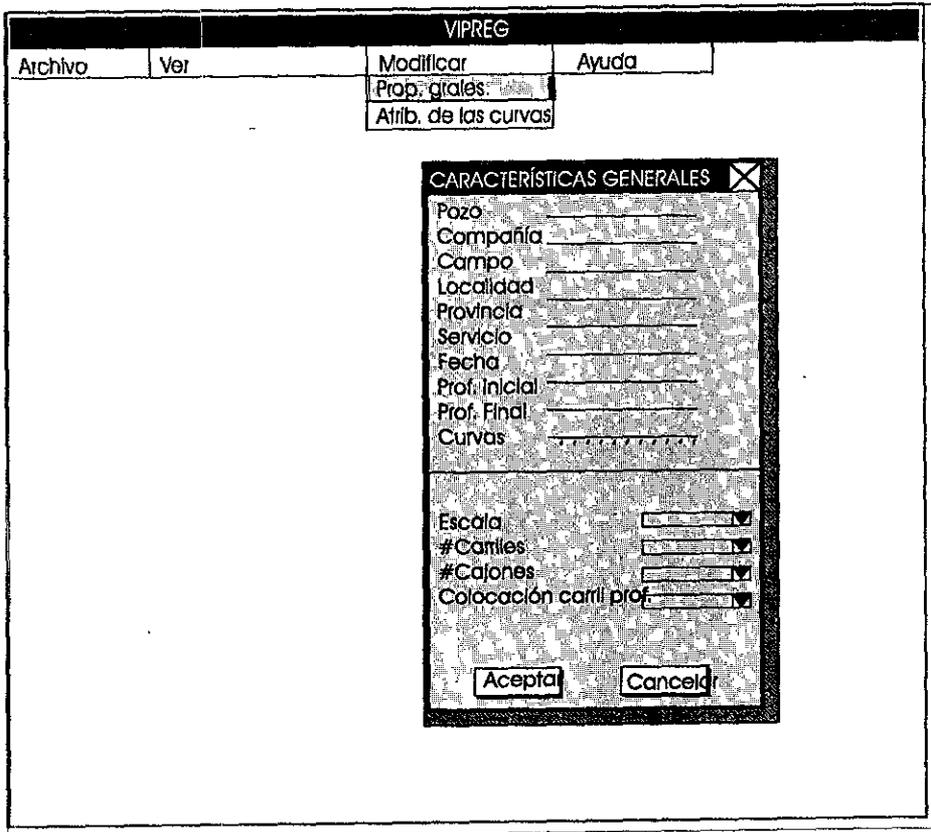


Ver Características Generales.

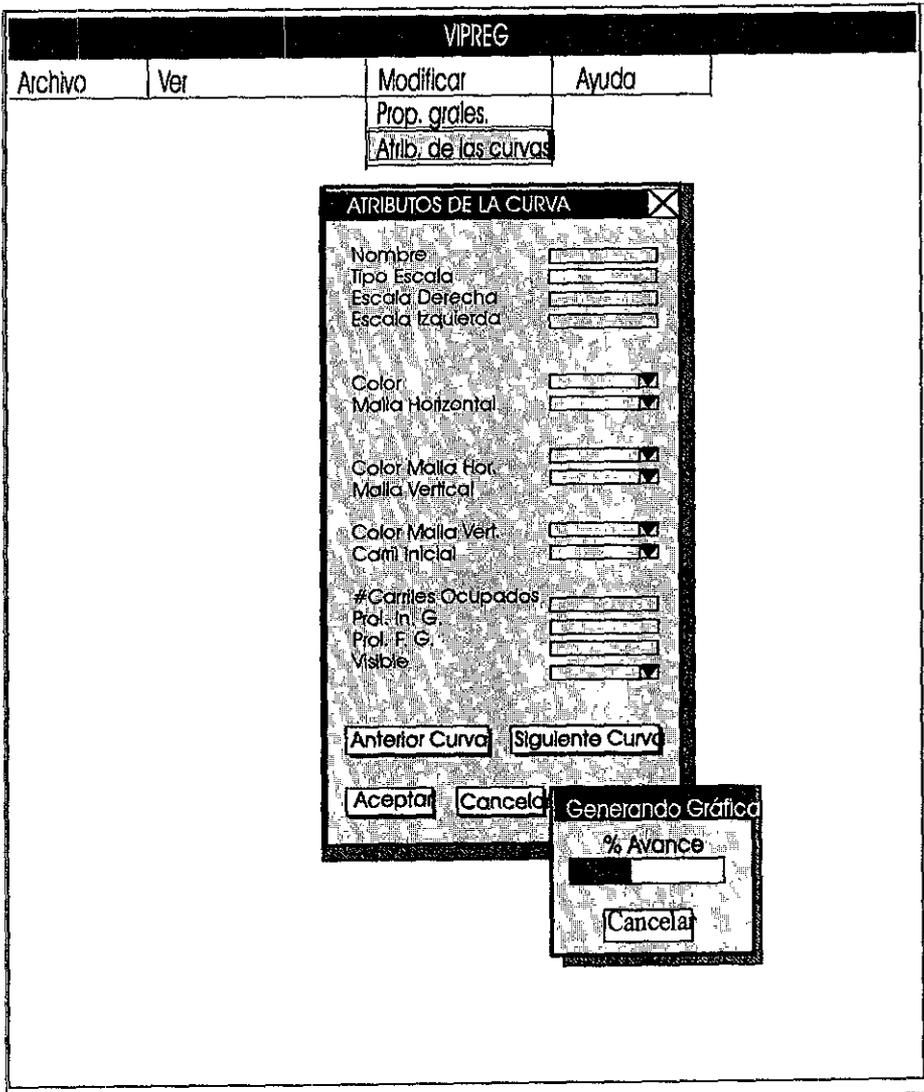
Diagramas de Interfaz



Diagramas de Interfaz

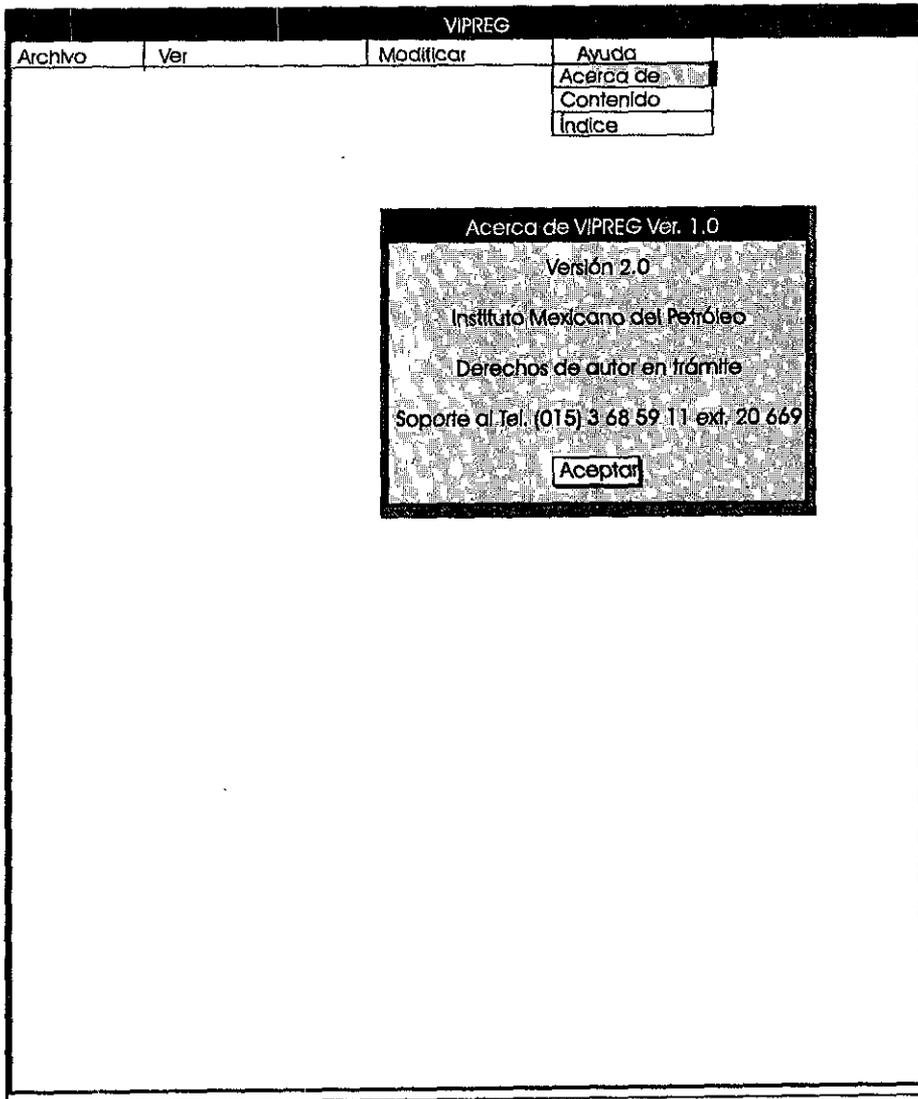


D11 Modificar Propiedades Generales.



D112 Modificar Atributos de las Curvas.

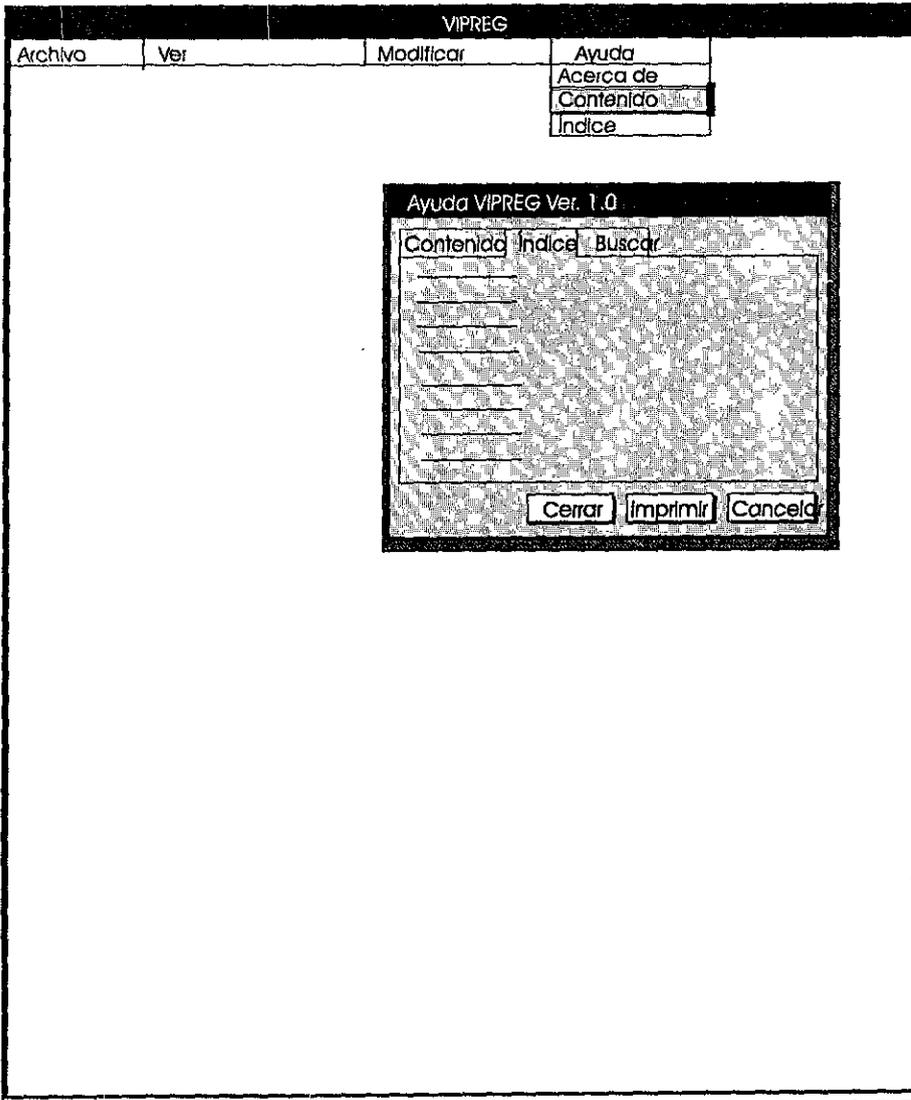
Diagramas de Interfaz



D113

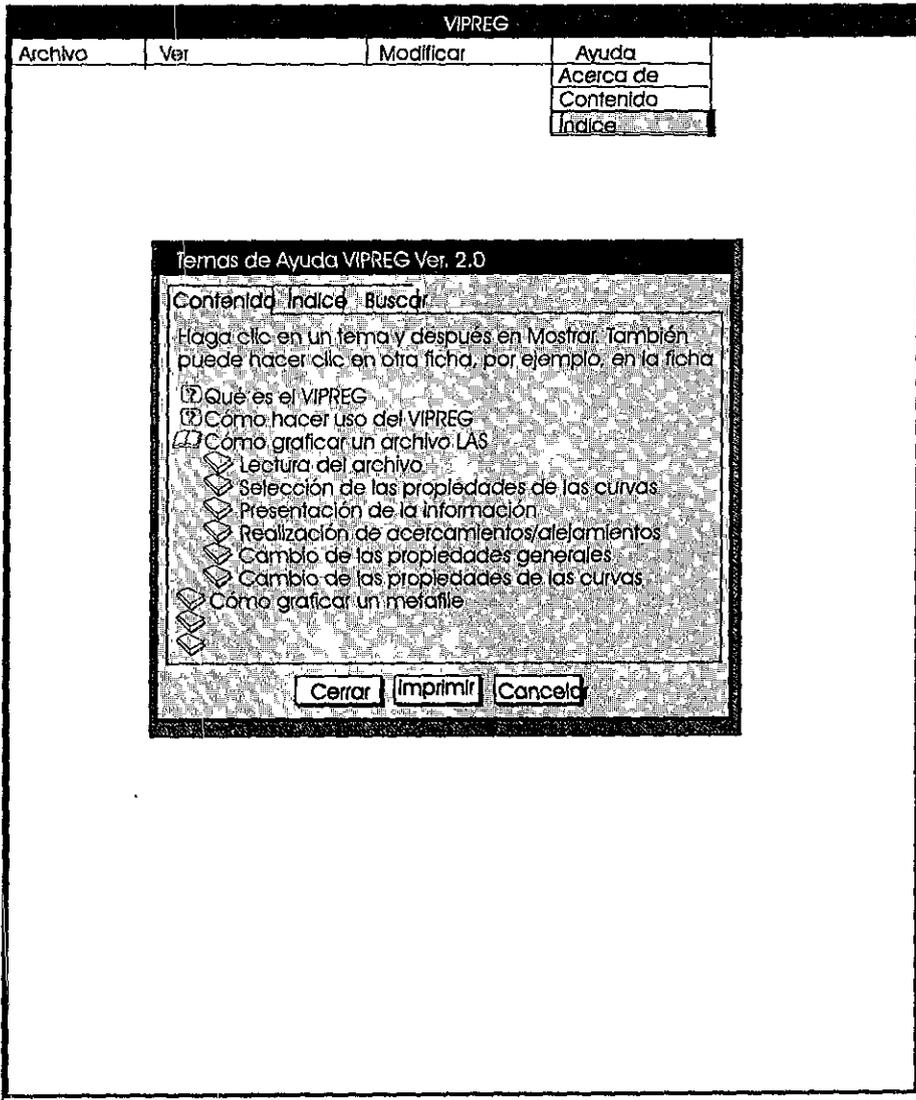
Ayuda Acerca De.

Diagramas de Interfaz



Di14 Ayuda Contenido.

Diagramas de Interfaz



D115

Ayuda Índice.

4.1.14 ANÁLISIS FUNCIONAL INTERNO

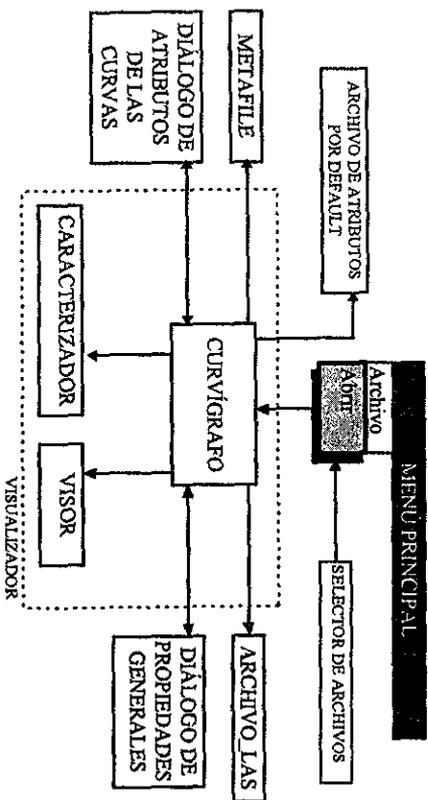
El análisis composicional definió la estructura preliminar del sistema y la funcionalidad externa o requerimientos fueron definidos con el análisis ambiental. El análisis funcional interno es definido por medio de trazas, cuya definición establece que *"Una traza es un camino específico de uso del sistema para realizar un segmento de funcionalidad, el cual identifica la interacción entre varios elementos para realizar una función específica"*, representando la liga o relación entre la estructura y los requerimientos del sistema.

El diagrama E11 del análisis funcional externo (Menú Principal) proporciona la base principal de identificación de trazas del sistema, los cuales se resumen en la tabla de la Fig.4.18

ID	NOMBRE	NEMÓNICO
DT1	Abrir Archivo de Datos LAS	TAAD
DT2	Abrir Metafile	TAM
DT3	Cerrar Archivo	TCA
DT4	Guardar Archivo	TGA
DT5	Guardar Archivo Como	TGAC
DT6	Salir	TSAL
DT7	Ver Barra Estándar	TVBE
DT8	Ver Barra de Herramientas	TVBH
DT9	Ver Propiedades Generales	TVPG
DT10	Ver Cajones	TVC
DT11	Ver Acercamiento/Alejamiento	TVAA
DT12	Modificar Propiedades Generales	TMPG
DT13	Modificar Atributos de las Curvas	TMAC
DT14	Ayuda Acerca de	TAACER
DT15	Ayuda Contenido	TACONT
DT16	Ayuda Índice	TAIND

Fig. 4.18. tabla de trazas del sistema VIPREG.

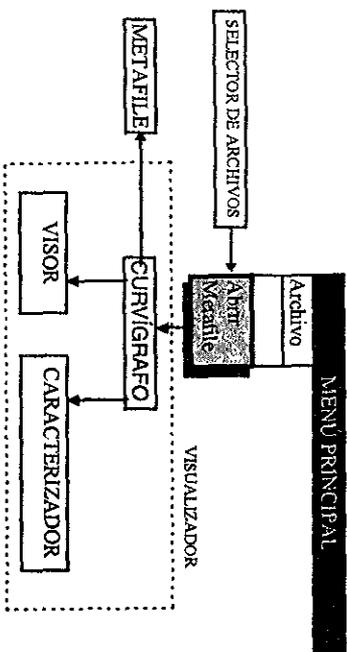
Análisis de Trazas



1. El usuario selecciona del menú principal la función **Abrir**.
2. Aparece el diálogo de selección del cual se elige un archivo residente en un directorio local.
3. CURVÍGRAFO recibe el nombre del archivo, lo decodifica y presenta las características generales y atributos de las curvas para modificación.
4. CURVÍGRAFO genera una gráfica de datos.
5. CURVÍGRAFO establece el proceso de presentación visual para el usuario inicial.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA".
Traza : Abrir Archivo de Datos TAAD ID : DT11
Elaboración : 30 de Sept.
Modificaciones : 16 de Abril de 1998

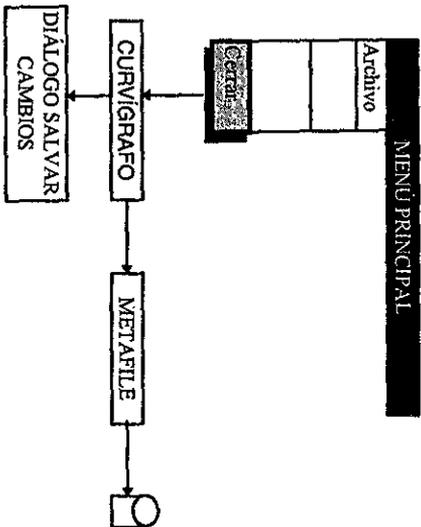
Analisis de Trazas



1. El usuario selecciona del menú principal la función **Abrir Metafile**.
2. Aparece el diálogo de selección del cual se elige un metafile.
3. CURVIGRAFO recibe el nombre del archivo, lo decodifica y presenta las características generales.
4. CURVIGRAFO lee el metafile, estableciendo el proceso de presentación visual para el usuario inicial.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA".
Traza : Abrir Metafile TAM ID : DT2
Elaboración : 30 de Sept.
Modificaciones : 6 de Abril de 1998.

Análisis de Trazas

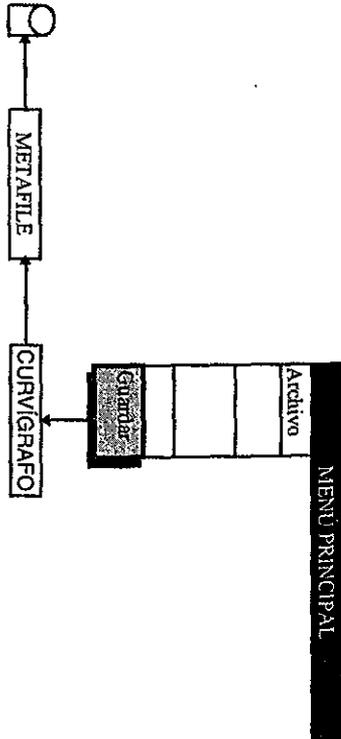


1. El usuario selecciona del menú principal la función Cerrar.
2. CURV/IGRAFO valida la existencia de cambios.
Si existen cambios :
 - Se presenta dialogo con opción de Salvar
 - Si usuario Salva
 - Se guardan los cambios realizados a Metafile.
 - De Otra manera
 - No se guardan los cambios.
 - De Otra manera
 - Se termina el proceso.
3. Se cierra el archivo actual.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Traza : Cerrar Archivo TCA ID : DT3
Elaboración : 30 de Sept.
Modificaciones : 6 de Abril de 1998.

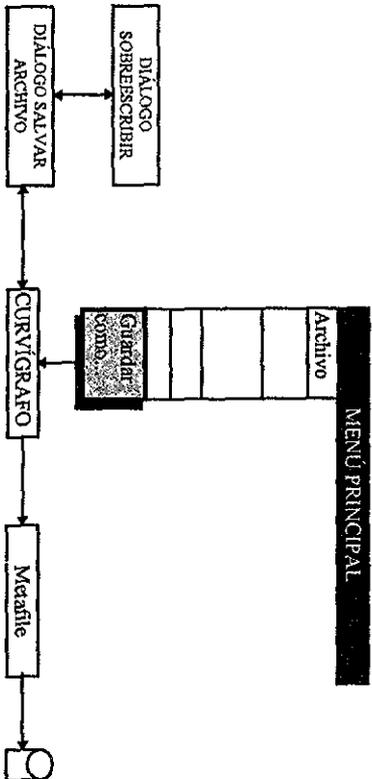
Analisis de Trazas

Browser gráfico para la visualización de la información petrolera. Un modelo básico.



1. El usuario selecciona del menú principal la función Guardar.
2. CURVIGRAFO escribe el archivo en el disco.
Si existen cambios
 - Se guarda Metafile.
 - De Otra manera
 - No se guarda el archivo, debido a que no existen cambios.
3. Se retorna el control a la aplicación para atender nuevos eventos.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA".
Trazas : Guardar Archivo TGA ID : DT4
Elaboración : 30 de Sept.
Modificaciones : 6de Abri de 1998.

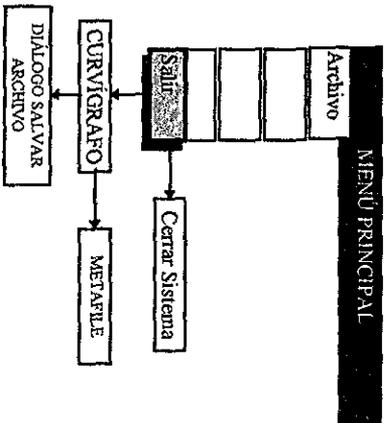


1. El usuario selecciona del menú principal la función Guardar Como.
2. Se presenta diálogo de salvar archivo como.
3. El usuario selecciona un nuevo nombre.
 - Si ya existe un archivo con ese nombre.
 - Se presenta diálogo con opción de sobrescribir el archivo
 - Si usuario acepta sobrescribir
 - Se sobrescribe el archivo con la nueva información.
- De Otra manera
 - Se retorna al paso 2.
- De Otra manera
 - Se guarda Metatfile con el nuevo nombre.
4. Se cierra el archivo anterior.
5. CURVIGRAFO presenta la información del archivo nuevo.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
 INFORMACIÓN PETROLERA"
 Taza : Guardar Archivo Como TGAC ID : DTS
 Elaboración : 30 de Sept.
 Modificaciones : 6 DE Abril de 1998.

Análisis de Trazas

Browser gráfico para la visualización de la información petrolera. Un modelo básico.



1. El usuario selecciona del menú principal la función Salir
2. CURVÍGRAFO válida la existencia de cambios.
 - Si existen cambios
 - Se presenta dialogo con opción de Salvar Si usuario Salva
 - Se guardan los cambios realizados a Metafile.De Otra manera
 - Si existen mas graficas abiertas se retorna al punto 2.
3. Se cierran todos los archivos actuales.
4. Sale de la aplicación.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".

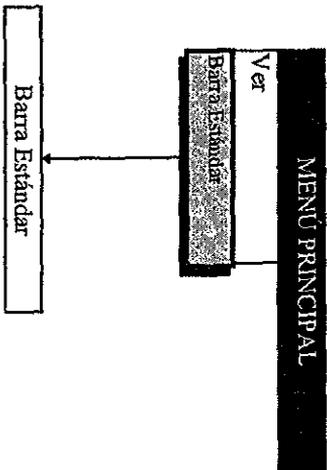
Traza : Salir TSAL

ID : DT6

Elaboración : 30-09-97;

Modificaciones : & de Abril de 1998.

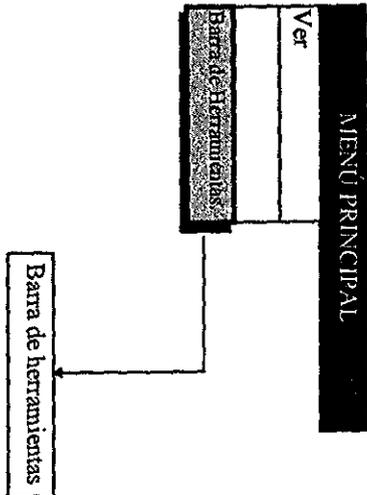
Analisis de Trazas



1. El usuario selecciona del menú principal la opción Ver.
2. El usuario selecciona del submenú la opción Barra Estándar
3. Se presenta/oculta la Barra Estándar.

SISTEMA "BROWSER GRAFICO PARA LA VISUALIZACION DE LA INFORMACION PETROLERA".
Trazas : Ver Barra de Herramientas TVBE ID: D17
Elaboración : 30-09-97 ;
Modificaciones : & de Abril de 1998.

Analisis de Trazas

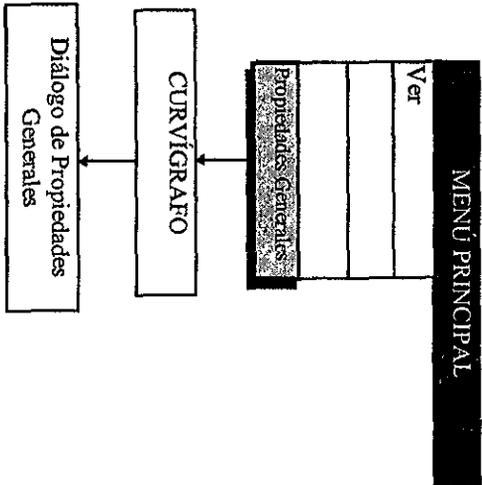


1. El usuario selecciona del menú principal la opción Ver.
2. El usuario selecciona del submenú la opción Barra de Herramientas.
3. Se presenta la Barra de Herramientas

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazo : Ver Barra de Herramientas TVBH ID : DT8
Elaboración : 30-09-97;
Modificaciones : & de Abril de 1998.

Análisis de Trazas

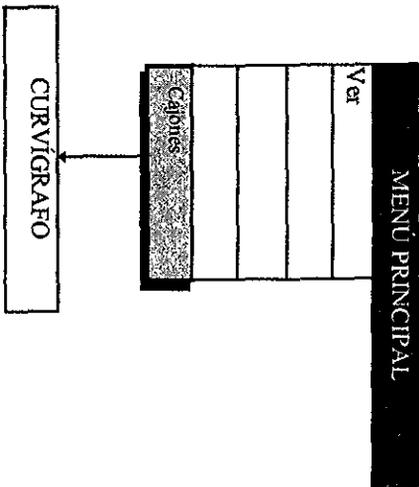
Browser gráfico para la visualización de la información petrolera. Un modelo básico.



1. El usuario selecciona del menú principal la opción **Ver**.
2. El usuario selecciona del submenú la opción **Propiedades Generales**.
3. Se presenta diálogo de **Propiedades Generales** de la gráfica actual.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA".
Trazas : Ver Propiedades Grales TVPG ID : D19
Elaboración : 30-09-97 ;
Modificaciones : 6 de Abril de 1998.

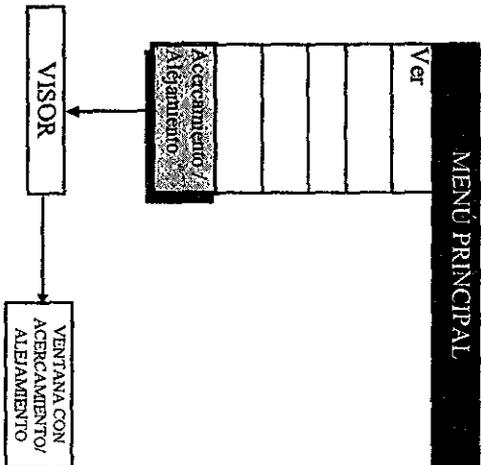
Análisis de Trazas



1. El usuario selecciona del menú principal la opción Ver.
2. El usuario selecciona del submenú la opción Ver Cajones.
3. Se presentan/ocultan los cajones de la gráfica.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas :Ver Arribatos Curvas TVC ID : D19
Elaboración : 30-09-97 ;
Modificaciones : 6 de Abril de 1998.

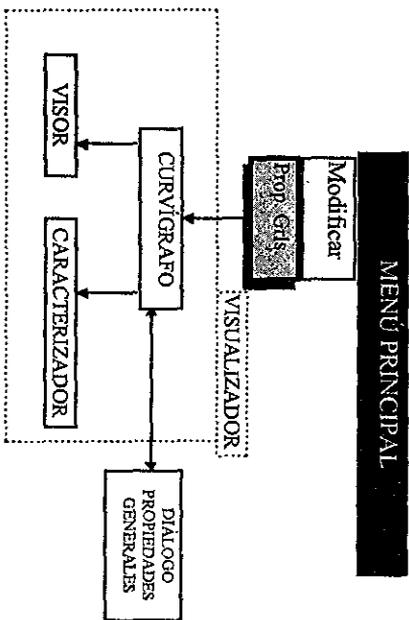
Analisis de Trazas



1. El usuario selecciona del menú principal la opción **Ver**.
2. Del submenú de **Ver** selecciona **Acercamiento/Alejamiento**.
3. El cursor se posiciona en el lugar inicial deseado dentro de la gráfica, se oprime el botón derecho del ratón y se arrastra hasta la posición final deseada liberándose el botón.
4. **VISOR** crea una nueva ventana insertando la información seleccionada con la opción **acercamiento/alejamiento**.
5. La nueva ventana se presenta y se detienen las demás partes del sistema hasta que ésta se cierre.
6. Se retorna el control al sistema.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas : Ver Acercamiento/Alejamiento TVAA ID : DT11
Elaboración : 30-09-97 ;
Modificaciones : 6 de Abril de 1998.

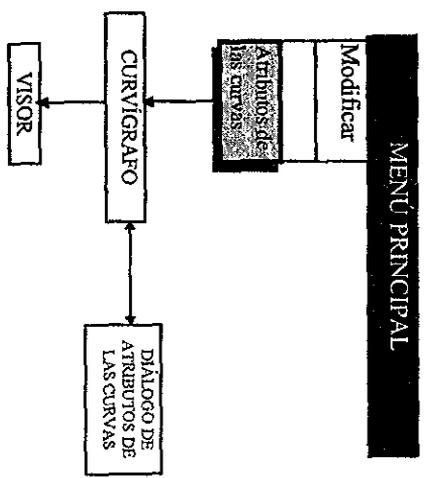
Analisis de Trazas



1. El usuario selecciona del menú principal la opción **Modificar**.
2. El usuario selecciona del submenú la opción **Propiedades Generales**.
3. Se presenta el diálogo para la modificación de las propiedades generales.
4. El usuario selecciona los cambios deseados.
5. Se verifican los cambios y son procesados por **CURVIGRAFO**.
6. **CURVIGRAFO** actualiza el **CARACTERIZADOR** y el **VISOR**.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA
INFORMACIÓN PETROLERA".
Trazas : Modifica Curvas TMPG ID : DT12
Elaboración : 30-09-97;
Modificaciones : 6 de Abril de 1998.

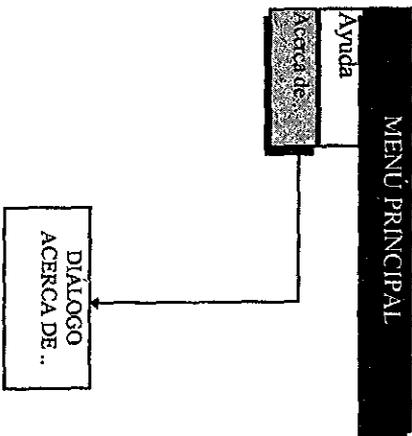
Analisis de Trazas



1. El usuario selecciona del menú principal la opción **Modificar**.
2. El usuario selecciona del submenú la opción **Atributos de las Curvas**.
3. Se presenta diálogo de modificación de los Atributos de las Curvas.
4. El usuario selecciona los cambios y son procesados por **CURVIGRAFO**.
5. Se verifican los cambios y son procesados por **CURVIGRAFO**.
6. Se actualiza el Visor y la información de los cajones.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas: Modifica Propiedades Grales. TMAC ID: DT13
Elaboración: 30-09-97;
Modificaciones: & de Abril de 1998.

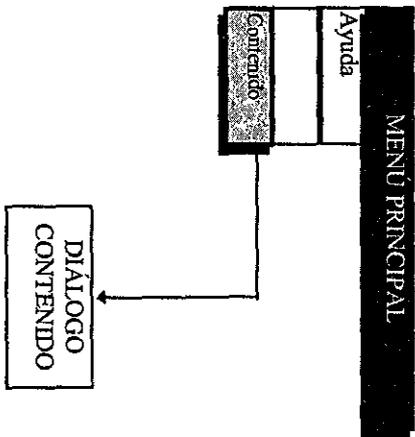
Analisis de Trazas



1. El usuario selecciona del menú principal la opción **Ayuda**.
2. Del submenú de **Ayuda** selecciona **Acercas de**.
3. Se presenta el diálogo con la opción seleccionada.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas : Ayuda Acerca de TAACER ID : DT14
Elaboración : 30-09-97 ;
Modificaciones :

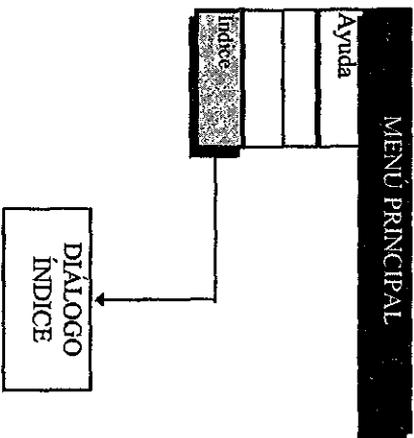
Análisis de Trazas



1. El usuario selecciona del menú principal la opción **Ayuda**.
2. Del submenú de **Ayuda** selecciona **Contenido**.
3. Se presenta el diálogo con la opción seleccionada.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas : Ayuda Contenido TACONT ID : DT15
Elaboración : 30-09-97 ;
Modificaciones :

Análisis de Trazas



1. El usuario selecciona del menú principal la opción **Ayuda**.
2. Del submenú de **Ayuda** selecciona **Índice**.
3. Se presenta el diálogo con la opción seleccionada.

SISTEMA "BROWSER GRÁFICO PARA LA VISUALIZACIÓN DE LA INFORMACIÓN PETROLERA".
Trazas :Ayuda Índice TALIND ID : DT16
Elaboración : 30-09-97 ;
Modificaciones :

4.2 DISEÑO DEL SISTEMA

El diseño tiene como propósito proporcionar la arquitectura de una estructura que soporte los requerimientos del análisis y su funcionalidad. El modelo de diseño del sistema es comúnmente representado por diagramas de clase y objetos, arquitectura y los mecanismos necesarios para manejo de errores, administración de memoria, almacenamiento de datos, etc. Finalmente el análisis y el diseño proveen un modelo o prototipo básico de desarrollo del sistema, el cual debe ser validado en el dominio de la aplicación.

Esta fase es muy parecida a la obtención del plano del diseño (diagrama) de un componente electrónico, donde cada línea es definida y se conocen los tiempos de conmutación de reloj, antes de ensamblar los elementos en las tarjetas de circuito impreso que corresponden a la fase de implementación.

El modelo de diseño requiere de :

- Obtención del modelo inicial de diseño.
- Identificar el ambiente de implantación.
- Describir la interacción entre los componentes del modelo.
- Realizar el diseño detallado de cada uno de los componentes.

4.2.1 CARACTERISTICAS GENERALES DE DISEÑO

Existen dos esquemas básicos para el diseño del visualizador, dependiendo de su forma de operación, el primero es el uso de una área de trabajo principal, la cual es conmutada para presentar una de las gráficas disponibles en la pila de gráficas (Fig. 4.17).

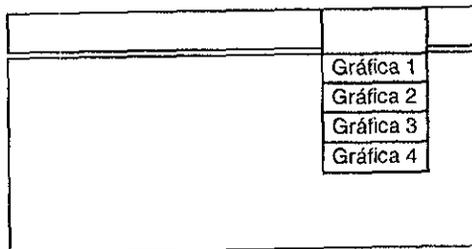


Fig. 4.17 Sistema con una ventana principal y una sola vista

El segundo consiste en el manejo de varias ventanas secundarias para las gráficas disponibles como se muestra en la Fig. 4.18

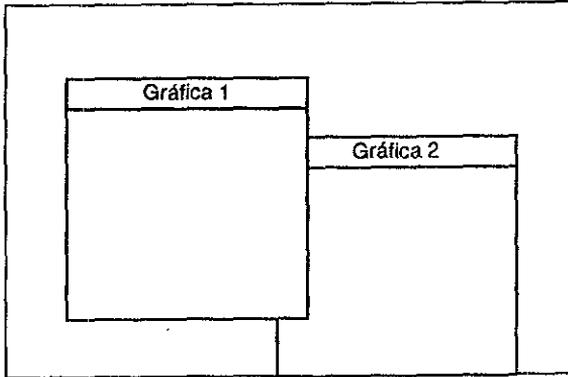


Fig 4.18 Sistema multiventanas

Cada uno posee cualidades y características de diseño diferentes. El primero permite un mayor control de los recursos disponibles, pero restringe la operabilidad del sistema, el segundo incrementa esta, pero requiere de un mayor número de recursos.

4.2.2 AMBIENTE DE DESARROLLO

El sistema se desarrollará bajo el sistema operativo Windows 95, utilizando un compilador que permita el desarrollo de aplicaciones con ventanas (*Windows*). La arquitectura típica de una aplicación en este ambiente es representada en la Fig. 4.19.

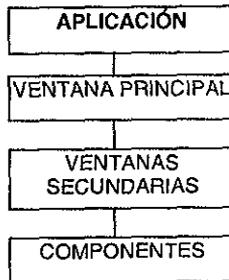


Fig. 4.19 Arquitectura básica de una aplicación en un ambiente de ventanas.

La aplicación define el contexto general de operación del sistema, la ventana principal es la ventana raíz sobre la cual se crea el menú principal del sistema y sirve como controlador principal. Las ventanas secundarias son derivadas de la principal con características predefinidas y sirven como base para la realización de las tareas del sistema.

4.2.3 MODELO INICIAL DEL SISTEMA

El modelo inicial permite establecer las condiciones y características generales de diseño del sistema, que sirven como lineamientos para el diseño detallado, además se describe un modelo básico con las relaciones que se perciben de los componentes, el cual será afinado durante toda esta fase.

La Fig. 4.20 muestra el modelo inicial del visualizador, de acuerdo con los resultados del análisis.

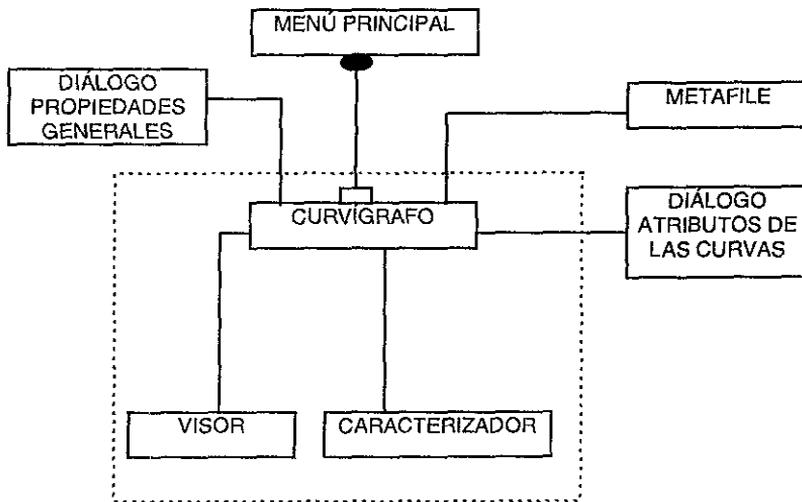


Fig. 4.20 Modelo inicial de diseño.

El modelo describe las relaciones existentes entre los componentes del sistema, las cuales en su mayoría son asociativas, lo que indica la existencia de comunicación por servicios solicitados entre componentes; la única relación de inclusión que puede definirse es la del MENÚ PRINCIPAL y el CURVÍGRAFO, ya que el primero define el número de gráficas que el sistema va a soportar, cuya propuesta son 10, dependiendo de la capacidad del equipo.

4.2.3.1 MENÚ PRINCIPAL

Esta representado por el elemento de interfaz EI1, el cual es ligado a la ventana principal, y funge como controlador principal del sistema ya que permite realizar el conjunto de operaciones definidas para el sistema. El diseño detallado debe establecer las relaciones y jerarquías de los elementos de interfaz, durante la implementación de los diagramas de interfaz, haciendo un análisis de sensibilidad con el número de gráficas generadas y el número de diálogos que va a manejar.

4.2.3.2 VISUALIZADOR

Es el componente que permite la interacción del sistema con el usuario, capturando todos los eventos y proporcionando la información disponible. Se divide en tres subcomponentes, cada uno abarcando un segmento de funcionalidad de acuerdo con los siguientes características :

1. CURVÍGRAFO

Este componente es quien controla toda la información de la gráfica que será desplegada en una ventana (principal o secundaria), y esta formado por los cajones de las curvas (Fig.4.21).

Los cajones de las curvas deben ser construidos como áreas activas o sensitivas a determinados eventos del ratón, que permitan modificar sus atributos en tiempo real.

2. CARACTERIZADOR

Administra y presenta las características de la gráfica de datos. El elemento de interfaz EI9 muestra los atributos más importantes de este componente, el cual es construido en base a un diálogo. El diseño detallado debe especificar el uso de un caracterizador para todo el sistema o un caracterizador para cada gráfica desplegada.

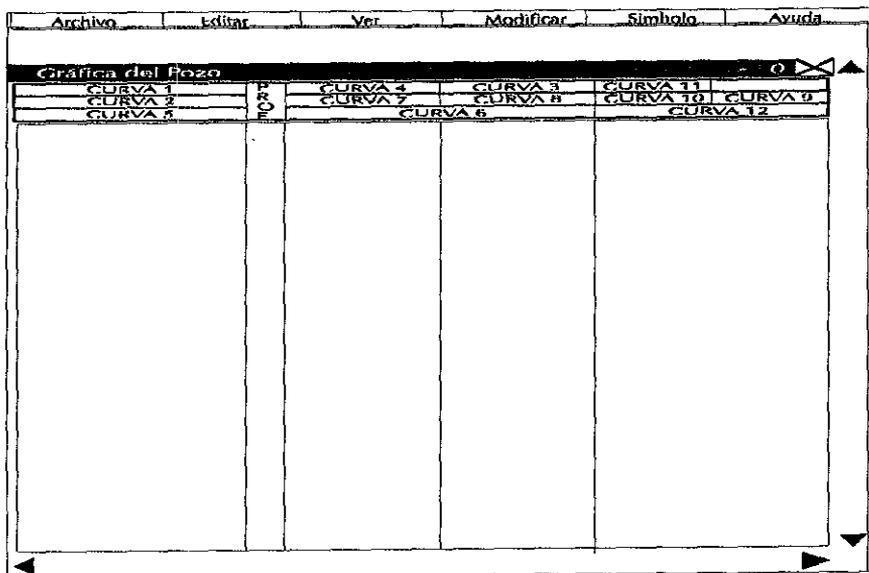


Fig.4.21. Elementos del CURVÍGRAFO.

3. VISOR

Permite la visualización de la gráfica de datos en una área de una ventana, la cual es controlada por dos barras deslizables (scrollbars), las que sirven para indicar posición y parte del segmento de gráfica visible. El diseño detallado de este subcomponente debe especificar las políticas de representación dentro del *viewport*, ya que una gráfica de más de 4mts. de largo requiere más de 20 *viewports*, siendo lento el recorrido a través de ésta.

Otra de las características de diseño del visor es la especificación del manejo de edición de símbolos y de los elementos que componen a la gráfica, ya que todo el *viewport* debe ser sensitivo a la captura de eventos del ratón o teclado.

Durante la fase de análisis el visualizador fue manejado por sus subcomponentes, lo cual permitió un mejor desarrollo de las trazas, de igual manera se manejara durante la fase de diseño funcional.

4.2.3.3 METAFILE

Representa la imagen de una gráfica de datos respaldada en disco, por lo que las características de diseño están orientadas al desarrollo de la estructura de un archivo que soporte el almacenamiento de los diferentes segmentos gráficos.

4.2.3.4 DIÁLOGO DE PROPIEDADES GENERALES

Tiene como función básica presentar las propiedades generales de la gráfica para su modificación.

4.2.3.5 DIÁLOGO DE ATRIBUTOS DE LAS CURVAS

Tiene como función principal la presentación de los atributos de las curvas para su modificación. La utilización de un componente de estos por gráfica o para toda la aplicación dependerá de la flexibilidad del compilador.

4.2.4 DISEÑO FUNCIONAL

Tiene como objetivo fundamental especificar la interacción entre los elementos o componentes del sistema, la cual es realizada a través de diagramas de interacción.

4.2.5 DIAGRAMAS DE INTERACCIÓN

El diagrama de interacción describe como cada uno de los elementos de la traza definida durante el análisis. Esto es similar al diseño de un componente electrónico, cuando se definen cada una de las señales entre diferentes elementos que lo constituyen y la lógica de conmutación de éstas.

El diagrama muestra como los objetos de una traza interactúan debido al envío de estímulos de uno a otro, además tienen como objetivo principal definir el protocolo de comunicación entre los objetos, la notación empleada por éstos se muestra en la Fig. 4. 22.

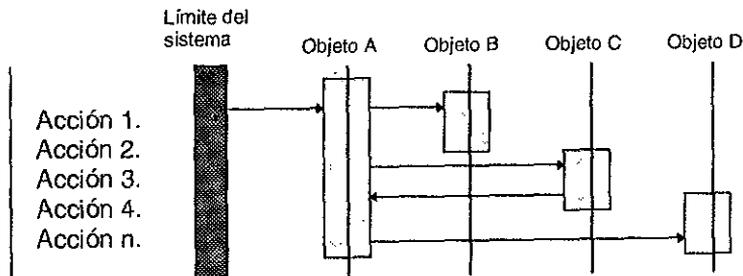


Fig. 4.22 Diagrama de Interacción.

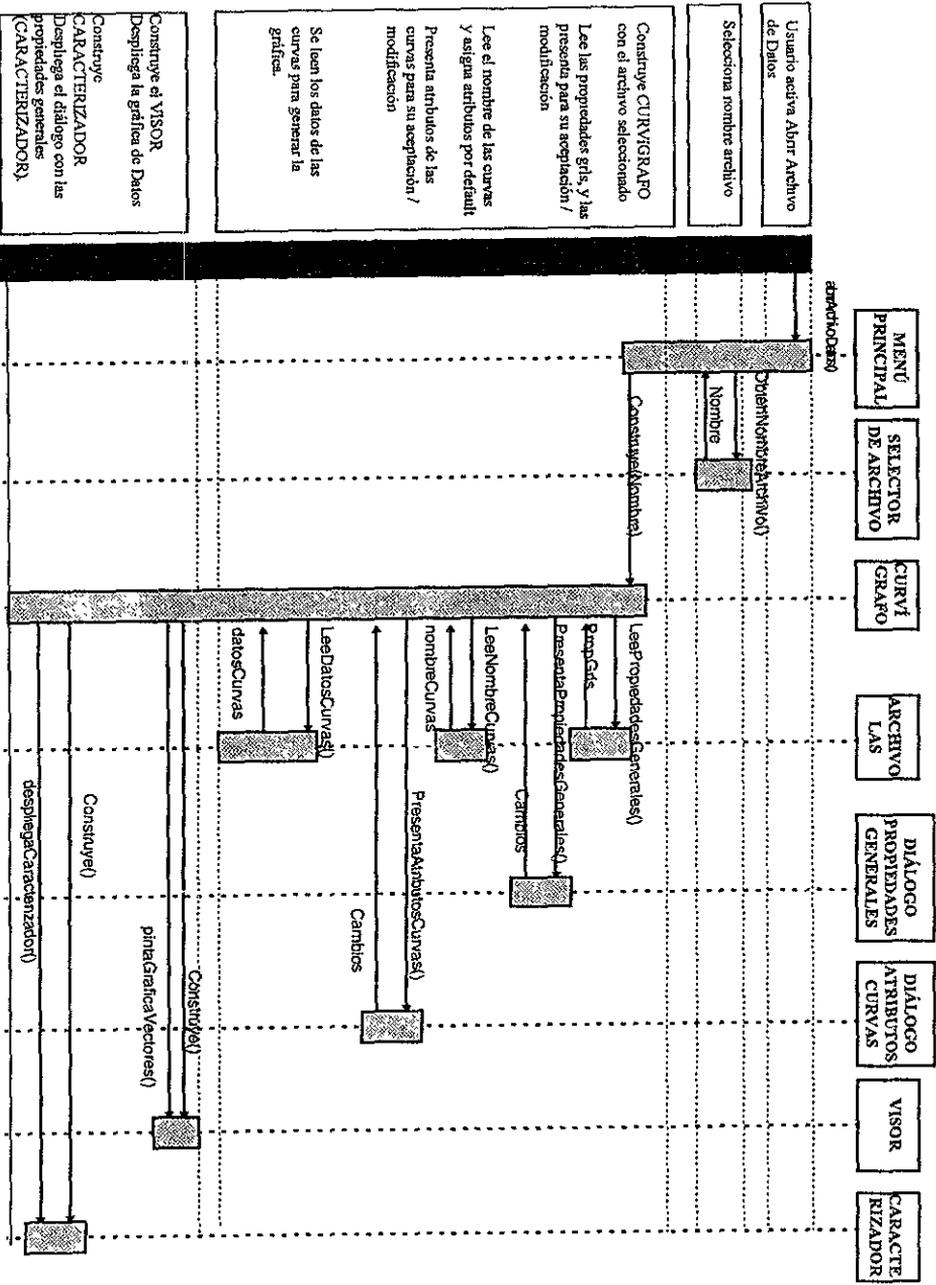
Cada objeto participante se representa con una barra, éstas son dibujadas como líneas verticales en el diagrama, el límite del sistema es representado por otra barra más gruesa, esta identifica la interfaz del sistema. En el lado izquierdo del límite del sistema se describen la secuencia de eventos de la traza que se está diseñando, la cual puede utilizar pseudocódigo o texto estructurado, éste describe que está pasando en esta parte de la traza con el objeto que está usando y se denomina operación. Las operaciones establecen servicios o funciones que los objetos realizan para la satisfacción del evento al cual pertenece la traza. La sincronización de operaciones se realiza a través del envío de estímulos o la petición de operación de un objeto a otro.

La tabla de la Fig. 4.23 muestra la relación de diagramas de interacción desarrollados para el browser, cuya numeración muestra su relación con los diagramas de traza.

ID	NOMBRE
DINT1	Abrir Archivo de Datos
DINT2	Abrir Metafile
DINT3	Cerrar Archivo
DINT4	Guardar Archivo
DINT5	Guardar Archivo Como
DINT6	Salir
DINT7	Ver Barra Estándar
DINT8	Ver Barra de Herramientas
DINT9	Ver Propiedades Generales
DINT10	Ver Cajones
DINT11	Ver Acercamiento/Alejamiento
DINT12	Modificar Propiedades Generales
DINT13	Modificar Atributos de las Curvas
DINT14	Ayuda Acerca de
DINT15	Ayuda Contenido
DINT16	Ayuda Índice

Fig. 4.23 Tabla de diagramas de interacción.

Diagramas de Interacción



DINT1.- Abrir Archivo de Datos.

Usuario actúa: Abre Archivo de Datos

Selecciona nombre archivo

Construye CURVIGRAFO con el archivo seleccionado

Lee las propiedades gis, y las presenta para su aceptación / modificación

Lee el nombre de las curvas y asigna atributos por default

Presenta atributos de las curvas para su aceptación / modificación

Se leen los datos de las curvas para generar la gráfica.

Construye el VISOR

Despliega la gráfica de Datos

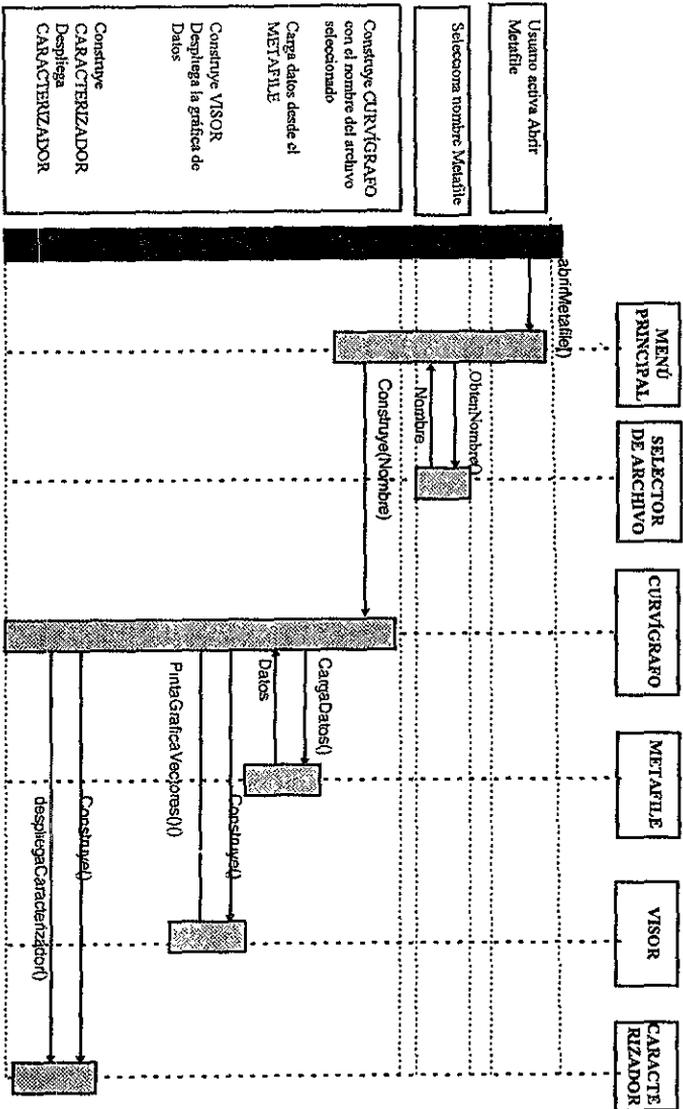
CARACTERIZADOR

Construye

Despliega el diálogo con las propiedades generales (CARACTERIZADORR)

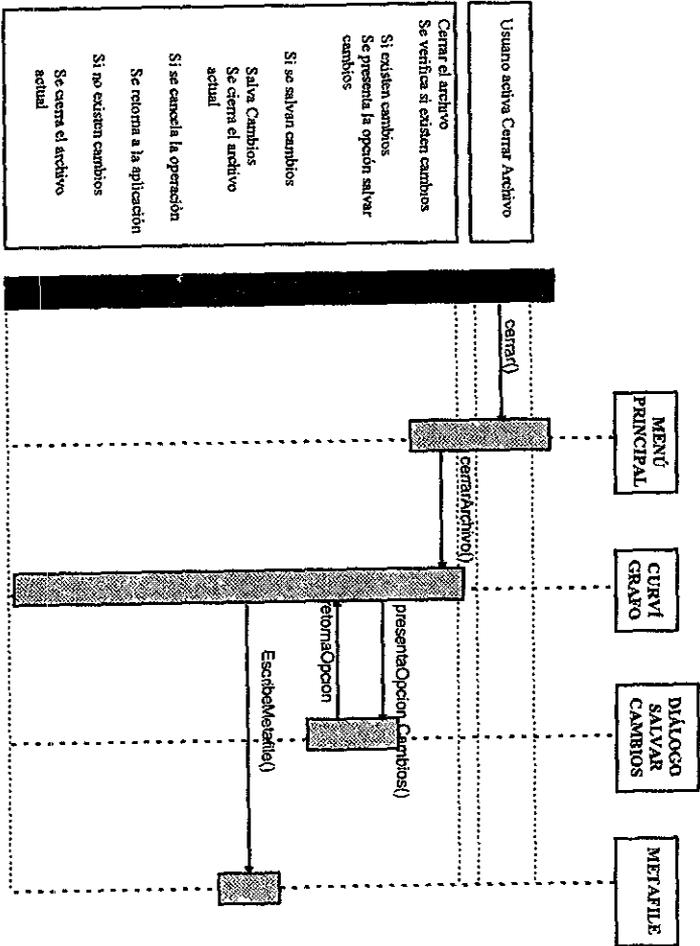
Diagramas de Interacción

Browseer gráfico para la visualización de la información petrolera. Un modelo básico.



DIN72.- Abrir Metafile.

Diagramas de Interacción



Usuario activa Cerrar Archivo

Cerrar el archivo
 Se verifica si existen cambios
 Si existen cambios
 Se presenta la opción salvar cambios

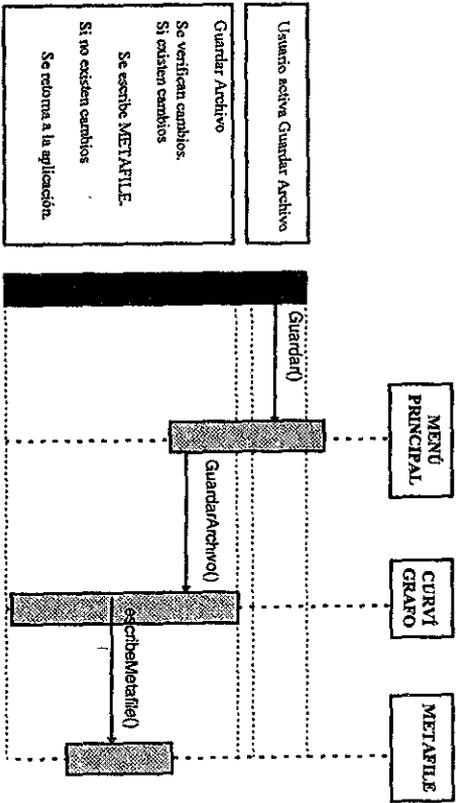
Si se salvan cambios
 Salva Cambios
 Se cierra el archivo actual

Si se cancela la operación
 Se retorna a la aplicación

Si no existen cambios
 Se cierra el archivo actual

DINT3.- Cerrar Archivo.

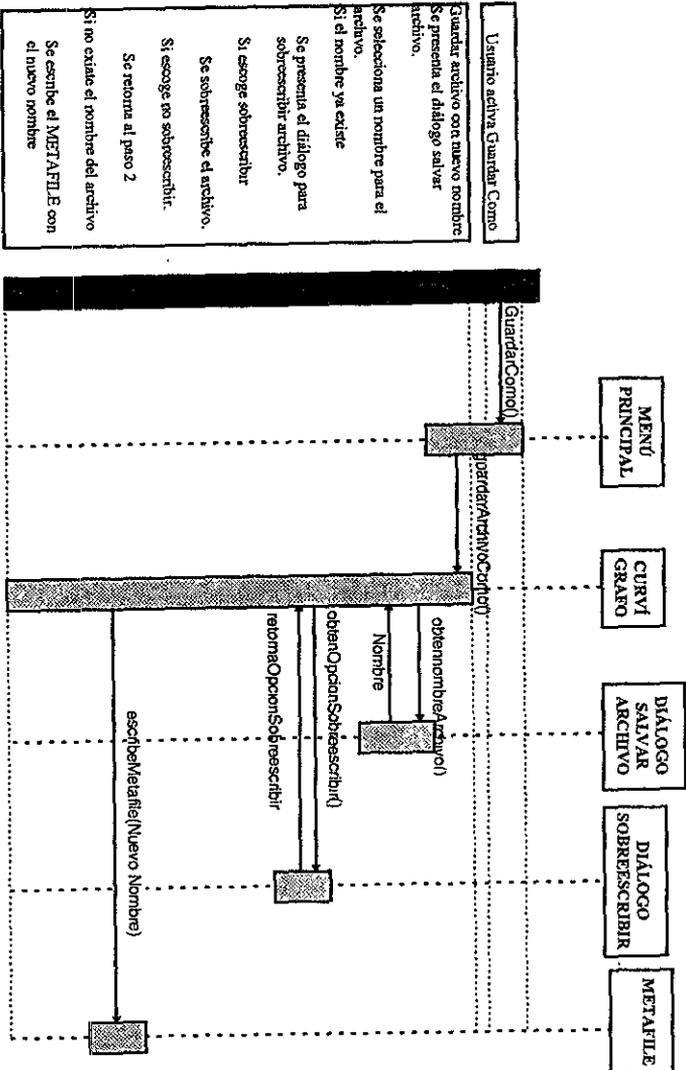
Diagramas de Interacción



DINTA.- Guardar Archivo.

Diagramas de Interacción

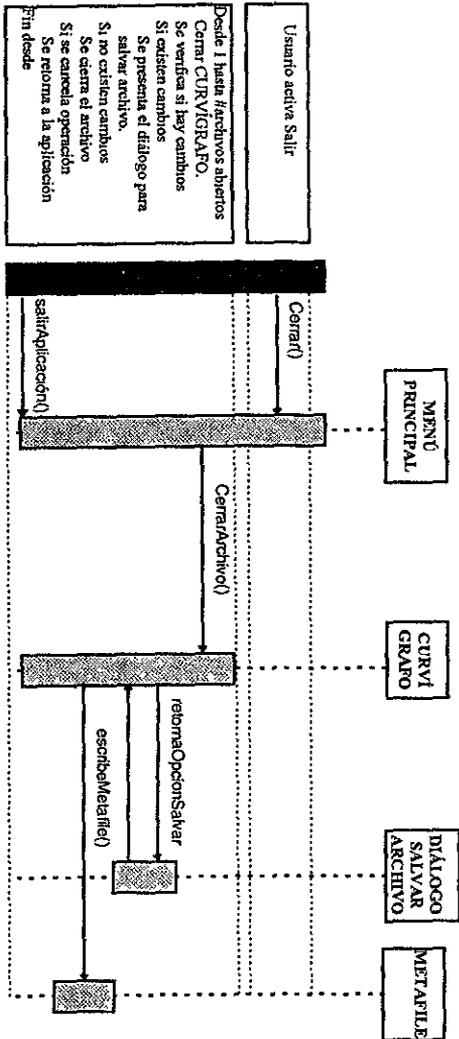
Browse gráfico para la visualización de la información petrologa. Un modelo básico.



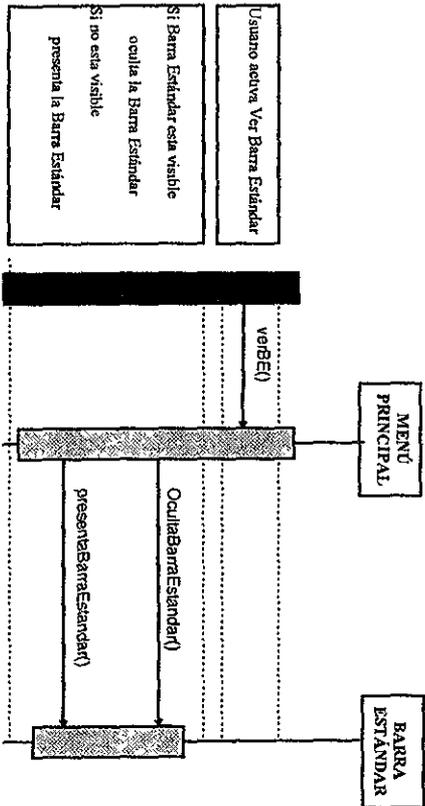
DINITS.- Guardar Como.

Diagramas de Interacción

Browser gráfico para la visualización de la información petrolera. Un modelo básico.

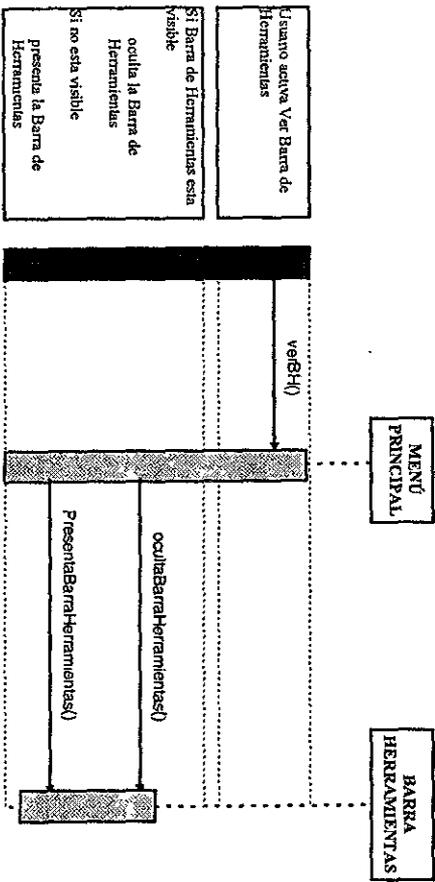


Diagramas de Interacción



DIN77.- Ver Barra Estándar.

Diagramas de Interacción



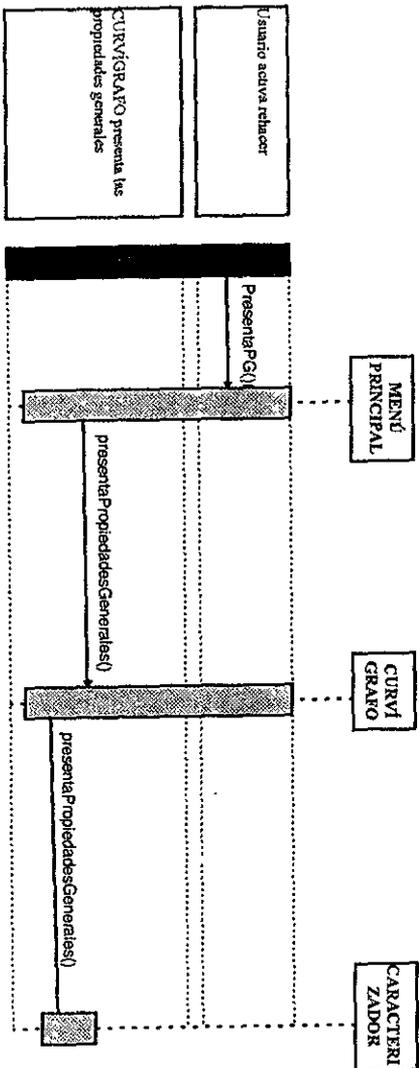
Usuario activa Ver Barra de Herramientas

Si Barra de Herramientas esta visible
oculta la Barra de Herramientas
Si no esta visible
presenta la Barra de Herramientas

DINT3.- Ver Barra de Herramientas.

Diagramas de Interacción

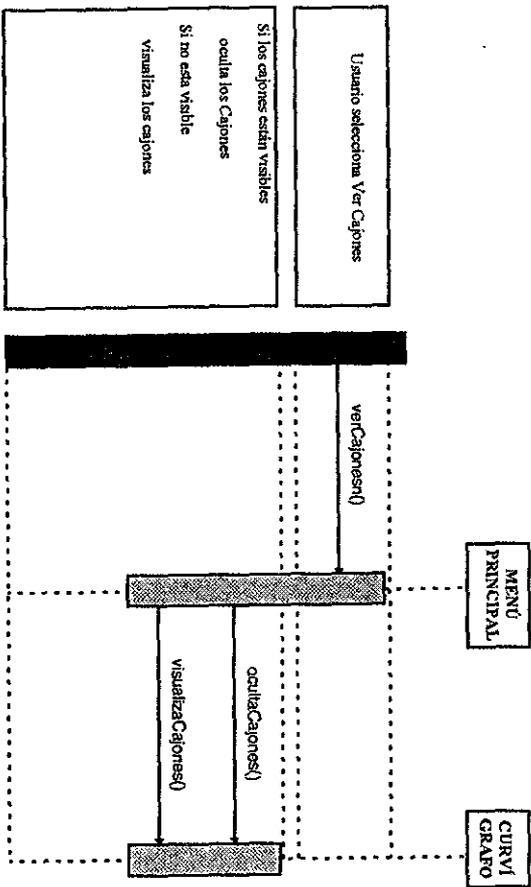
Browser gráfico para la visualización de la información petrolera. Un modelo básico.



DIN19.- Ver Propiedades Generales (Caracterizador).

Diagramas de Interacción

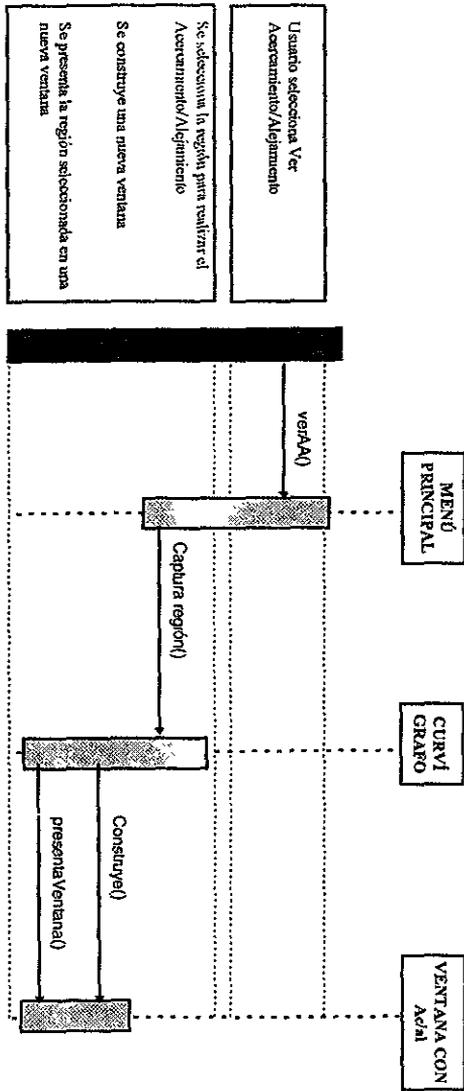
Browser gráfico para la visualización de la información petrologera. Un modo clásico.



DINTT0.- Ver Cajones.

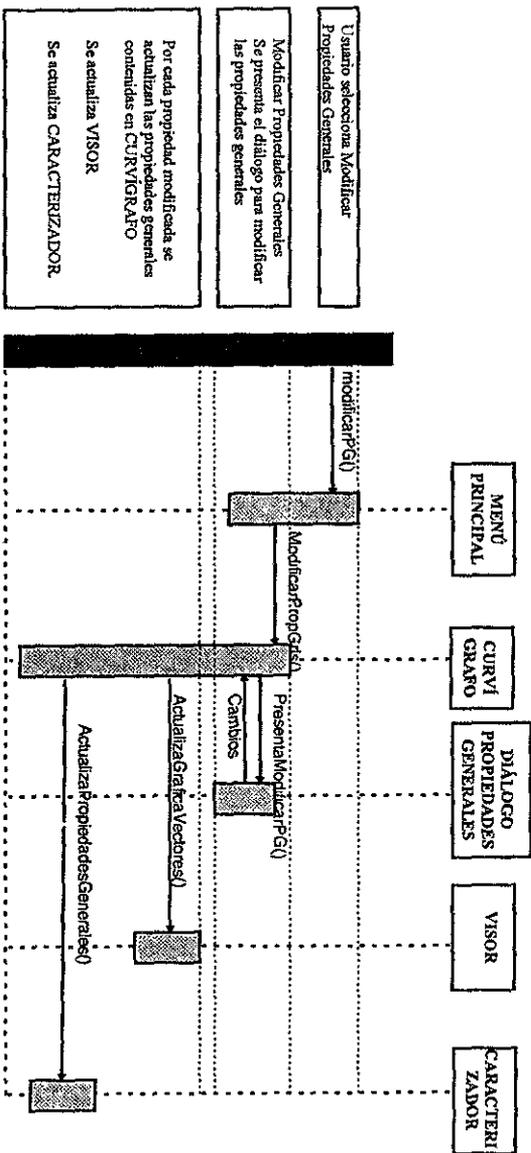
Diagramas de Interacción

Browser gráfico para la visualización de la información petrolera. Un modelo básico.



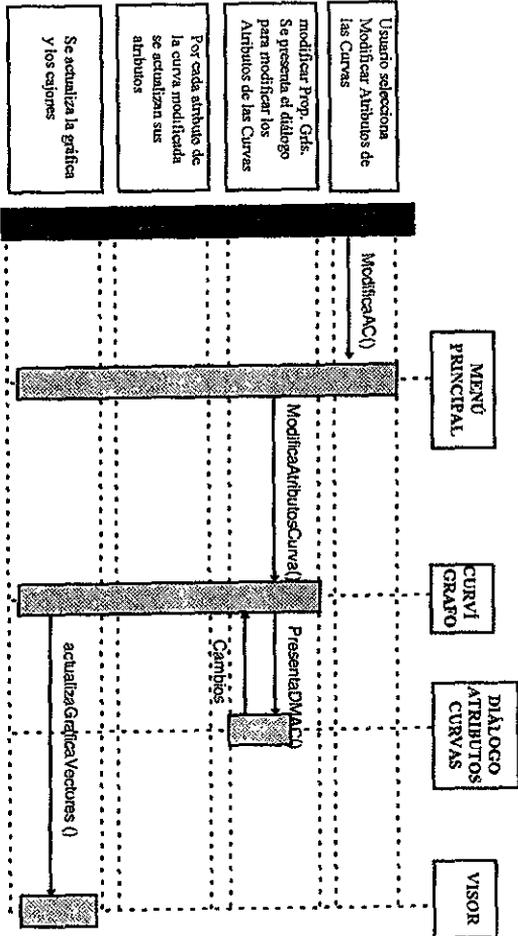
DINTT1.- Ver Acercamiento/Alejamiento.

Diagramas de Interacción



DINT12.- Modificar Propiedades Generales.

Diagramas de Interacción



Usuario selecciona Modificar Atributos de las Curvas

modificar Prop. Gte. Se presenta el dialogo para modificar los Atributos de las Curvas

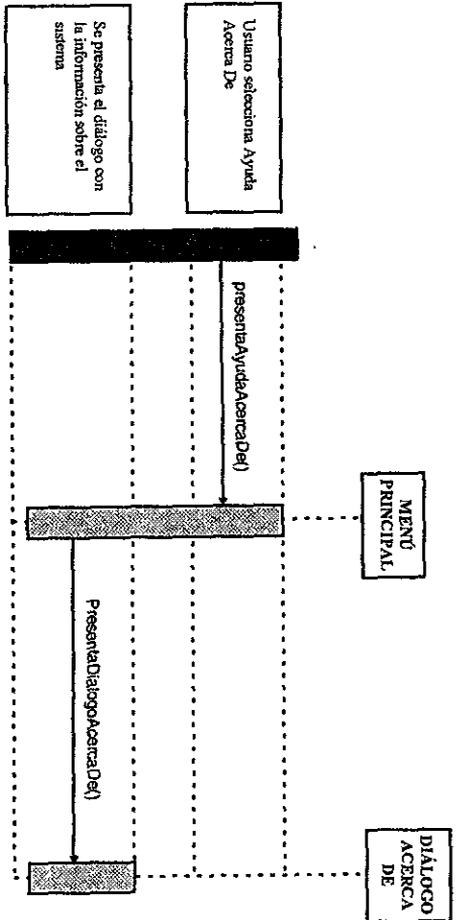
Por cada atributo de la curva modificada se actualizan sus atributos

Se actualiza la grafica y los ejtores

DINT13 Modificar Atributos de las Curvas.

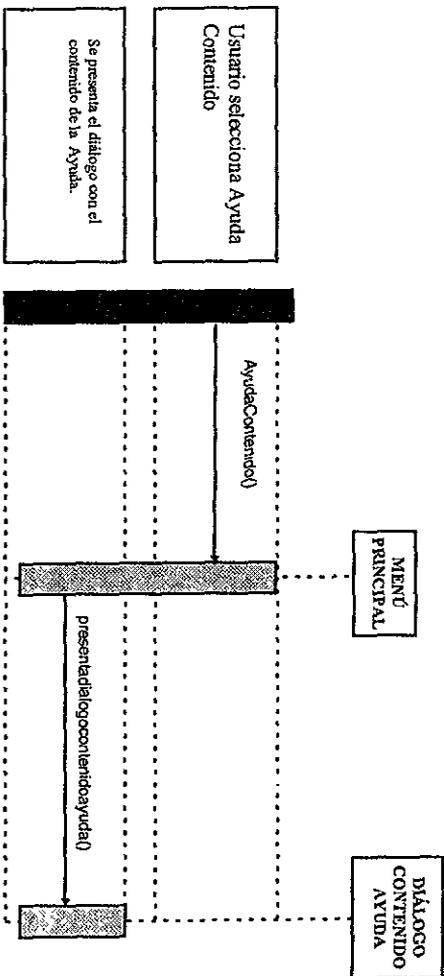
Diagramas de Interacción

Brower gráfico para la visualización de la información petrotera. Un modelo básico.



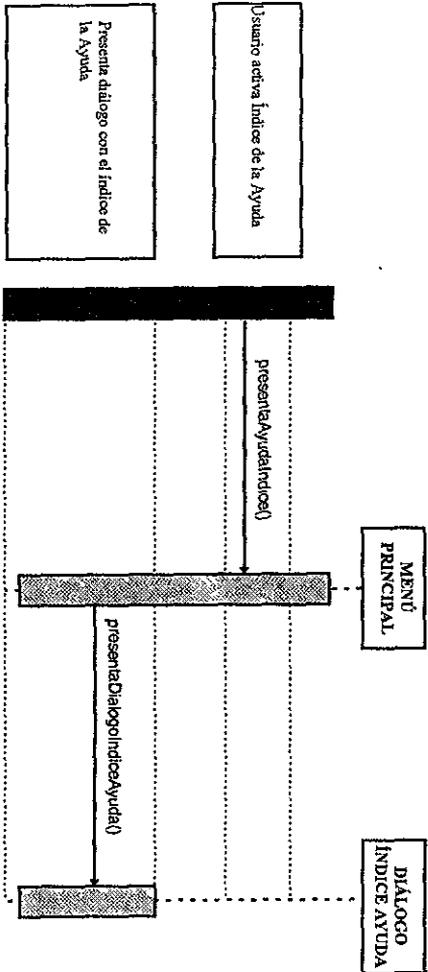
DINT14 Ayuda Acerca De.

Diagramas de Interacción



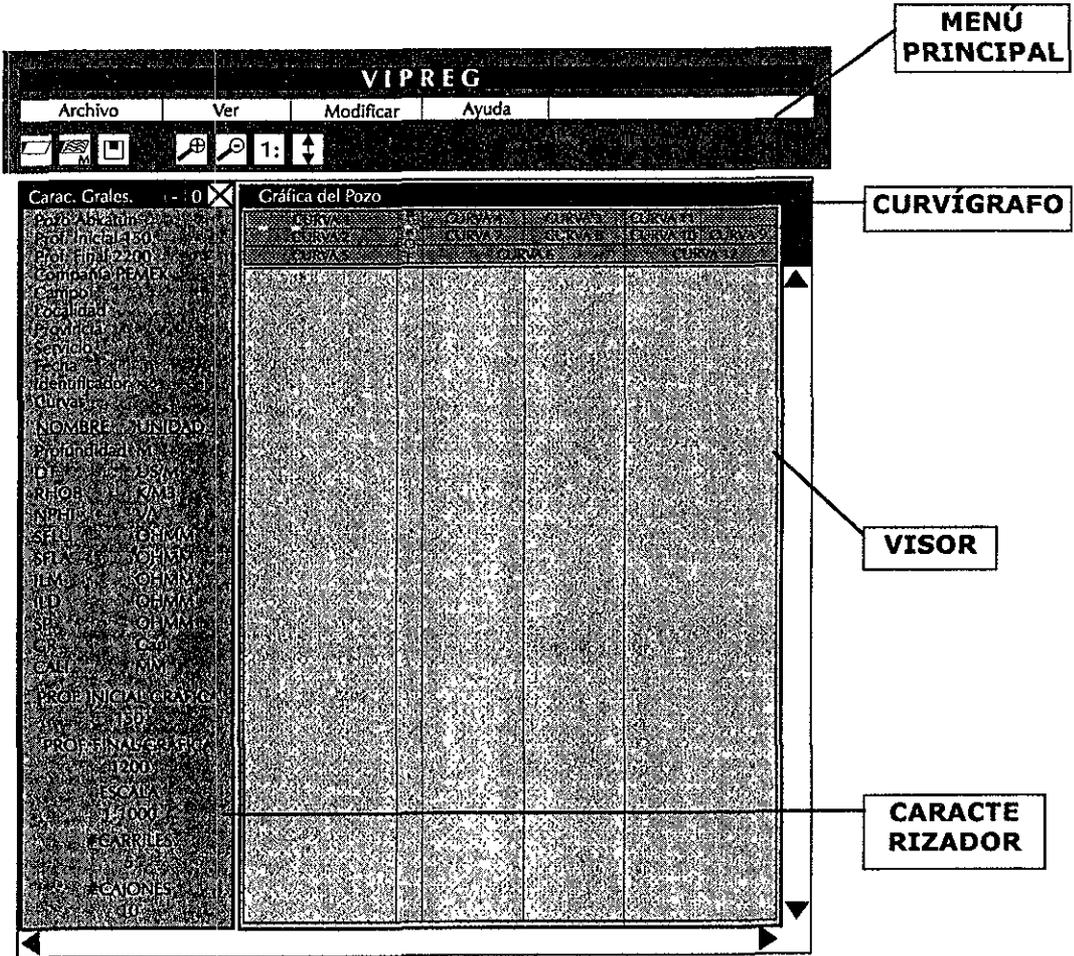
DINT15.-Ayuda Contenido.

Diagramas de Interacción



DINT16.- Ayuda Índice.

5. CONSTRUCCIÓN DEL MODELO BÁSICO



Visualizador de Información Petrolera. Registros Geofiscos
VIPREG



5.1 Especificación del browser

El *browser* gráfico para la visualización de la información petrolera que cumple con las características especificadas dentro de los requerimientos (punto 4.1.4), es especificado mediante la figura 5.1

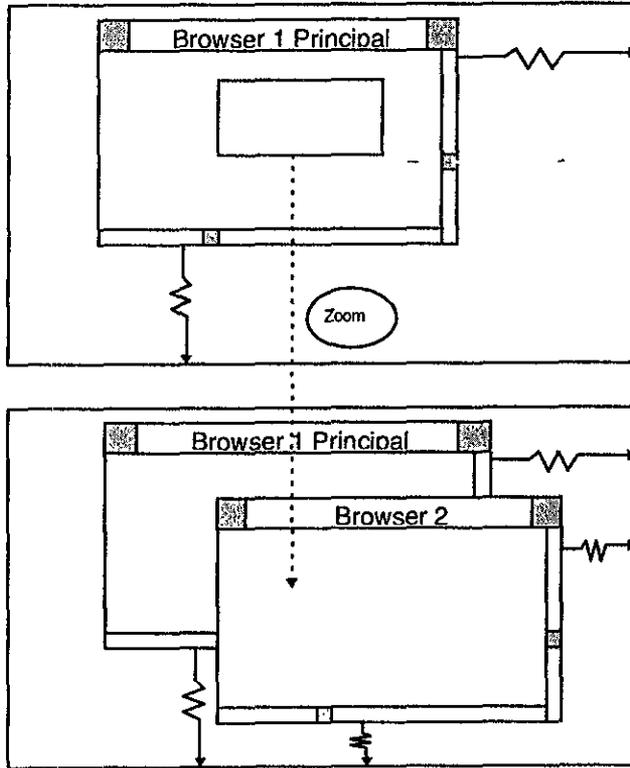
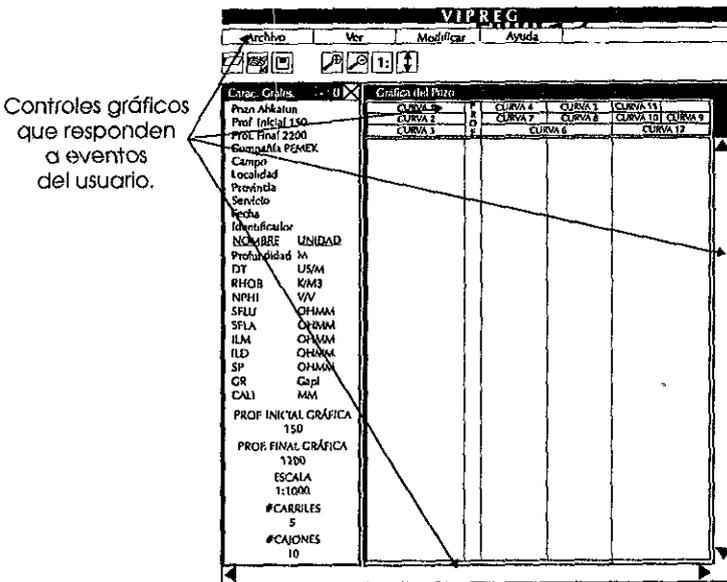


Fig. 5.1. Especificación del browser para la visualización de registros geofísicos de pozo.

Se trata de un browser de dos dimensiones con múltiples superposiciones (varias graficas) y libre zoom, dinámico, debido a que el usuario es libre de mover, colocar redimensionar y ocultar las ventanas en el lugar que el considere mas conveniente, genera una nueva ventana a partir de un campo visual seleccionado por medio de una operación "*Mouse down, drag, Mouse up*", es decir, el usuario elige dos puntos con el mouse y se genera un acercamiento/alejamiento, la cual es independiente de la primera ventana, por lo que no existe coordinacion entre ambas y el usuario tiene dos browsers con magnificaciones diferentes.

Todos los componentes del sistema están compuestos por controles gráficos que dependen completamente del compilador y la plataforma –Sistema Operativo- sobre el cual trabaja el sistema.

Los controles gráficos responden a eventos activados por el usuario y dependiendo del evento se llevan a cabo ciertos procesos y/o actividades predeterminadas, como lo muestra la figura 5.2.



En lenguajes de programación que trabajan sobre un sistema operativo Windows 95, existen clases predeterminadas por el compilador que cumplen con las necesidades para la implementación de controles gráficos. El nombre de las clases de cada uno de los objetos de este tipo dependen del compilador.

Para el diseño detallado de todos los componentes del VISUALIZADOR se explotará al máximo el uso de clases propias del compilador para optimizar al máximo la implementación del sistema. Cada control gráfico además de llevar consigo información sobre sus propiedades y funciones para su manejo contendrá información de interés - atributos de la curva, propiedades de la gráfica combinados con clases y estructuras de datos - de acuerdo a los procesos que se deseen ejecutar.

En este caso se analizaron 3 compiladores de C++ para Windows:

- Builder C++.
- Visual C++ y
- Borland C++.

Para lo cual se obtuvieron los siguientes resultados:

Builder C++⁵ tiene una analogía con el estilo de programación de Visual Basic. A partir de una "forma" se va integrando cada menú, diálogo, botones, etc., realizando la liga de todos y cada uno de los elementos con sus funciones de manera automática. Por ejemplo, si se crea una forma con un menú, seleccionando un elemento del menú, automáticamente "aparece" -se crea- una función dentro de la cual se escribe el código que se realiza al activar dicho elemento del menú. En otras palabras, ofrece un ambiente de desarrollo más agradable y sencillo con toda la potencia que cualquier compilador de C++ pueda tener.

En Borland C++ para la elaboración de un menú es necesario recurrir a otra aplicación llamada "Resource Workshop"⁶ y posteriormente hacer la liga "manual" de cada uno de los elementos por medio de programación, lo que representa confusión e incrementa el tiempo de programación y por consiguiente el costo del sistema.

En Visual C++⁷ la liga de los objetos del menú se realizan de forma similar a Borland C++. Se crea el menú, diálogo, botón, etc. y a cada uno de los componentes se les asigna un identificador, posteriormente se realiza la liga de cada uno de los elementos con la correspondiente función por medio de programación.

Por las facilidades para la codificación que presenta Builder C++, es sobre este compilador que se definirán todas las clases y/o controles gráficos que se emplean en el sistema.

Diseño detallado del browser.

El diseño detallado tiene como objetivo especificar las características -metaclases, clases, estructuras, métodos, operaciones, etc.- de cada uno de los componentes que forman el sistema.

5.2 Menú Principal

El Menú principal proporciona toda la gama de opciones que pueden ser utilizadas dentro del sistema: abrir un archivo LAS, abrir un metafile, modificar los atributos de las curvas, modificar las propiedades generales de la gráfica, presentar la ayuda, etc.

La mayoría de los compiladores desarrollados para trabajar sobre un sistema operativo Windows permiten el manejo de menús sin ningún problema.

Dentro del menú principal existen "aceleradores", los cuales son comandos por combinaciones de teclas (Alt + "A", Alt + "M") que permiten realizar acciones de una forma más rápida. También existen los que se conocen como "Gadgets" -botones aceleradores-, los cuales son pequeños iconos sensibles que generalmente se encuentran bajo el menú principal.

Los Gadgets al igual que los aceleradores permiten realizar acciones ó comandos dentro del sistema de una forma más rápida, debido a que evitan acceder continuamente el menú y después los submenús.

Para la implementación de un menú, cada uno de los compiladores de C++ para Windows poseen una jerarquía de clases que contienen todas las funciones y comandos para manejar un menú con aceleradores y Gadgets.

Así que este subcomponente solo necesita especificar el nombre de cada uno de los elementos que lo conforman.

Los requerimientos del sistema definen la posibilidad de visualizar varias gráficas simultáneamente, conocido como una aplicación MDI (multiventanas). Por lo cual es necesario determinar la forma más óptima para el manejo de las ventanas dentro del espacio de la pantalla.

MODO PRELIMINAR DEL MENÚ PRINCIPAL

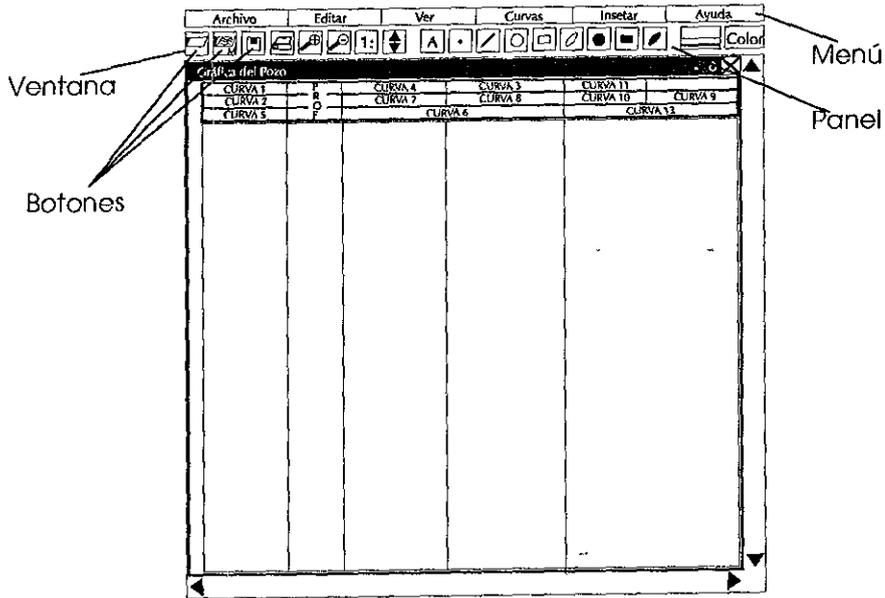
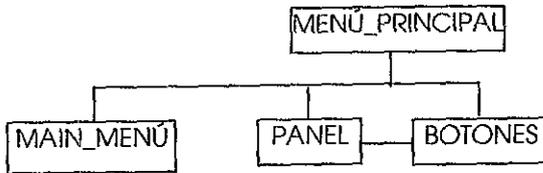


Fig. 5.3 . Partes que forman el MENU_PRINCIPAL.



Definición de los elementos

MENÚ_PRINCIPAL.- Controla la aplicación y funge como ventana principal.

MAIN_MENÚ.- Contiene las opciones del sistema, como abrir un archivo, guardar un archivo, etc.

PANEL_DE_BOTONES.- Es la región donde se despliegan los botones.

BOTONES.- Son botones aceleradores -Gadgets- para realizar las principales opciones de MAIN_MENU.

El objetivo de este componente es proveer al sistema un mecanismo de comunicación con el usuario para controlar y ejecutar las acciones o procesos.

A continuación se definen las funciones necesarias para que MENÚ_PRINCIPAL cumpla con todos sus requerimientos. Cabe señalar que en este caso fueron aprovechados al máximo los recursos propios del compilador, para realizar la programación más sencilla de los elementos como: menús, botones, aceleradores, diálogos propios del compilador y del sistema, etc.

**DEFINICIÓN DE LAS CLASES Y MÉTODOS DEL COMPONENTE
MENÚ PRINCIPAL**

class MENÚ_PRINCIPAL public TForm
métodos y funciones

Opción archivo

	Tipo	Actividad
abrirArchivoLas ()	TMenúItem	Abre un archivo Las y a partir de el genera su gráfica.
abrirMetafile ()	TMenúItem	Abre un metafile y presenta su gráfica.
cerrarArchivo ()	TMenúItem	Cierra el archivo actual.
guardarArchivo ()	TMenúItem	Guarda el archivo actual.
guardarArchivoComo ()	TMenúItem	Guarda el archivo actual con un nuevo nombre.
salirdeAplicación ()	TMenúItem	Salte de la aplicación.

Opción Ver

	Tipo	Actividad
verBarra_Estandar()	TMenúItem	Presenta/oculta la barra estándar.
verBarraDeHerramientas ()	TMenúItem	Presenta/oculta la barra de herramientas.
verPropiedadesGenerales ()	TMenúItem	Presenta el dialogo con las prop. Grls. de la gráfica actual.
verCajones ()	TMenúItem	Presenta/oculta los cajones de la gráfica actual..

Opción modificar

	Tipo	Actividad
modificarPropiedadesGenerales ()	TMenúItem	Presenta el diálogo para modificar las propiedades generales de la gráfica actual
modificarAtributosDeLasCurvas ()	TMenúItem	Presenta el diálogo para modificar los atributos de las curvas de la gráfica actual.

Opción Ventana

	Tipo	Actividad
presentarVentanasEncascada ()	TMenúItem	Presenta las ventanas en forma de cascada.
presentarVentanasEnMosaicoHoriz ()	TMenúItem	Presenta las ventanas en forma de mosaico vertical.
presentarVentanasEnMosaicoVert ()	TMenúItem	Presenta las ventanas en forma de mosaico Horizontal.
ordenarVentanasMinimizadas ()	TMenúItem	Ordena los iconos -ventanas minimizadas-.
minimizarTodasLasVentanas ()	TMenúItem	Minimiza todas las ventanas.

Opción Ayuda

	6. Tipo	Valor que regresa
presentarAcercaDe ()	TMenúItem	Presenta un diálogo con la información del sistema.
presentarContenidoAyuda ()	TMenúItem	Presenta los temas de ayuda.
presentarÍndiceAyuda ()	TMenúItem	Presenta el índice de la ayuda.

Gadgets ó aceleradores. Botones que permiten acceder rápidamente las funciones del menú.

	Clase	Función que ejecuta
BotónAbrirArchivoLas	TSpeedButton	abrirArchivoLas
BotónAbrirMetafile	TSpeedButton	abrirMetafile
BotónGuardarArchivo	TSpeedButton	guardarArchivo
BotónCortar	TSpeedButton	Cortar
BotónCopiar	TSpeedButton	Copiar
BotónPegar	TSpeedButton	Pegar
BotónAcercamiento	TSpeedButton	Acercamiento
BotónAlejamiento	TSpeedButton	Alejamiento
BotónCambiarEscala	TSpeedButton	CambiarEscala
BotónIrAProfundidad	TSpeedButton	IrAProfundidad
PanelDeBotones	Tpanel	Ninguna

CLASES PERTENECIENTES AL LENGUAJE DE LAS CUALES SE AUXILIA MENÚ_PRINCIPAL

	clase	
ABRIR ARCHIVO LAS	TOpenDialog	Abrir un Archivo - filtro *.LAS y *.* -
DIALOGO SALVAR	TSaveDialog	Guardar un Archivo
ABRIR METAFIJE	TOpenDialog	Abrir un Archivo - filtro *.MTF y *.* -
MAIN MENU	TMainMenu	Crear un menú dentro de la ventana

5.3 CARACTERIZADOR



Fig. 5.4. CARACTERIZADOR.

Caracterizador es un diálogo (fig. 5.4) formado con la información general de la gráfica; no existe interacción directa entre este componente con el usuario, ya que cuando modifica las características generales, lo realiza desde el menú principal, con la opción del menú Modificar y seleccionando el submenú Propiedades Generales.

En este caso, caracterizador es una imagen que presenta las características generales de la gráfica de una forma estática y esta construido a partir de una estructura de datos leída o proporcionada por CURVÍGRAFO, Estos datos deben ser validados y presentados en una ventana que podrá ser desplegada u ocultada desde el VISOR por medio de un botón, o desde el menú principal con la opción Ver, seleccionando el submenú Características Generales como se muestra en la figura 5.5. Además los datos presentados en el CARACTERIZADOR pueden ser actualizados en tiempo de ejecución.

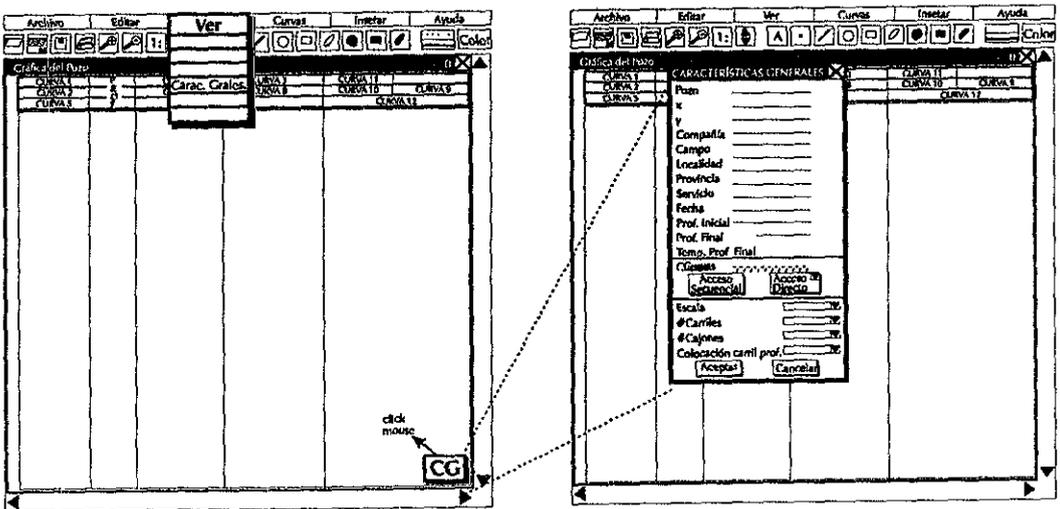


Fig. 5.5. Formas de presentar el CARACTERIZADOR.

MODELO PRELIMINAR DEL CARACTERIZADOR

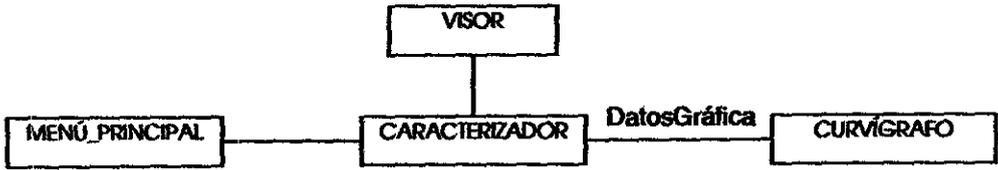
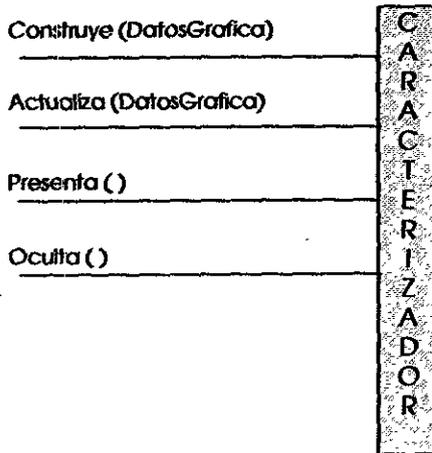


Figura 5.6. Modelo preliminar del caracterizador.

Requerimientos del caracterizador



DEFINICIÓN DE LAS CLASES Y MÉTODOS DEL COMPONENTE

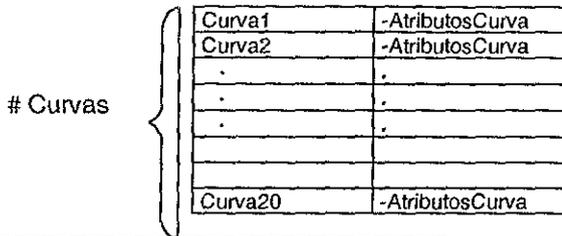
CARACTERIZADOR

class CARACTERIZADOR public TForm

Datos (atributos) necesarios para construir el CARACTERIZADOR

struct	-AtributosCurva
	Tipo
Nombre	String;
Unidad	String;
EscalalZquierda	double
EscalaDerecha	double
CambioEscalalZquierda	double
CambioEscalaDerecha	double
TipoEscala	int
TipoDeLinea	int
ColorCurva	int
MallaHorizontal	int
ColorMallaHorizontal	int
MallaVertical	int
ColorMallaVertical	int
CarrilInicial	int
NumeroCarrilesOcupados	int
Cañon	int
IntervaloProfundidadInicial	double
IntervaloProfundidadFinal	double
Visible	bool

struct	DatosGrafica
	Tipo
NombreDelPozo	AnsiString
ProfundidadInicial	double
ProfundidadFinal	double
Incremento	double
ValorNulo	double
Compañía	AnsiString
Campo	AnsiString
Localidad	AnsiString
Provincia	AnsiString
Servicio	AnsiString
Fecha	AnsiString
Identificador	AnsiString
NumeroDeCurvas;	int



ProfundidadInicialGrafica	double
ProfundidadFinalGrafica	double
Escala	int
NumeroDeCarriles	int
NumeroDeCajones	int
PosicionCarrilProfundidad	int

Para llevar el manejo y control de este componente son necesarios los siguientes métodos o funciones miembro.

Método	Actividad
Construye (DatosGrafica)	Realiza todos los procesos necesarios para crear el CARACTERIZADOR almacenando y conservando todas las propiedades generales de la gráfica.
ActualizaPropiedadesGenerales(Datos Grafica)	Actualiza todos los valores de las propiedades generales de la gráfica.
Presenta ()	Hace visible al caracterizador.
Oculto ()	Minimiza u oculta al caracterizador.
Crealetreros ()	Crealetreros que aparecen en el caracterizador.

5.4 CURVÍGRAFO

El curvígrafo es una región compuesta por pequeñas partes sensibles que accesan por medio de un "click del mouse", los atributos de cada una de las curvas de la gráfica para su consulta. Cada región sensible es conocida como cajón y sólo puede pertenecer a una curva.

La figura 5.7 muestra el escenario posible al aplicar un click con el mouse sobre la parte sensible que corresponde a una curva.

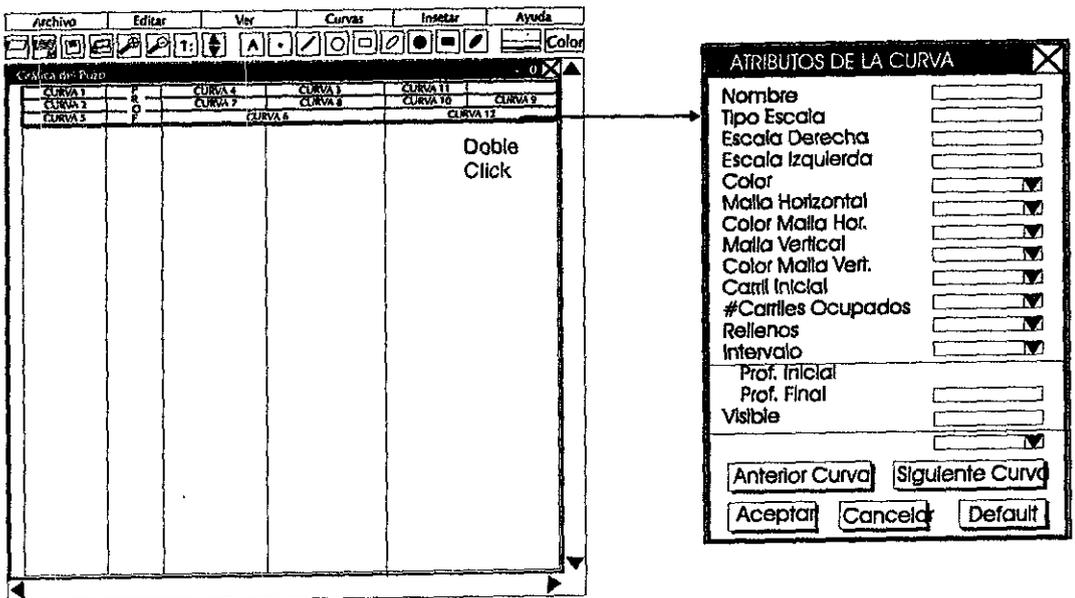


Figura 5.7. Acceso A los atributos de las curvas.

El usuario puede modificar cada uno de los atributos de la curva para adaptarla a sus necesidades.

Debido a que el carril de profundidad no posee el mismo tamaño de los carriles de las curvas, es necesario que las regiones sensibles (cajones qu forman el CURVÍGRAFO), sean objetos remodificables y/o reconstruibles en tiempo de ejecución, de tal forma que cuando se desee cambiar de lugar el carril de profundidad (como lo especifican los requerimientos), cada uno de los regiones sensibles sean reacomodables y redimensionables en tiempo real. Figura 5.8.

1	2	3	4
SP	P	RG	NPHI
NC	R	CAL	DT
NL	O	RHDB	LLD
	F		

1	2	3	4
SP	RG	P	NPHI
NC	CAL	R	DT
NL	RHDB	O	LLD
		F	

Figura 5.8. Reorganización de los cajones de la gráfica.

Cada cajón que forma parte del curvígrafo debe contener la información que presenta la figura 5.9. Por lo que para su construcción CURVÍGRAFO recibirá estos datos y los transformará en la presentación necesaria.

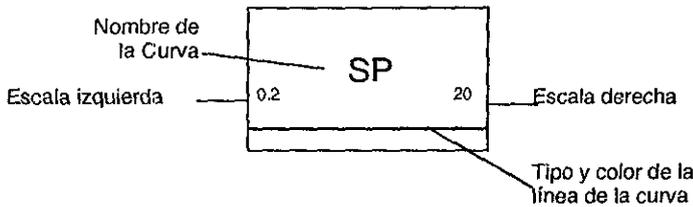


Figura 5.9. elementos de un cajón

El CURVÍGRAFO está formado por un conjunto de $n \times m$ elementos `_CajónCurva` donde n representa el número de cajones y m el número de carriles que contiene la gráfica. Para la creación del CURVÍGRAFO hay que considerar que una curva puede ocupar el mismo carril que otras curvas, pero NO el mismo cajón, es decir, un cajón sólo puede ser ocupado por una y sólo una curva. El número máximo de carriles es 10 (sin considerar el carril de profundidad) y el número máximo de cajones es 5. Fig. 5.10.

CURVÍGRAFO

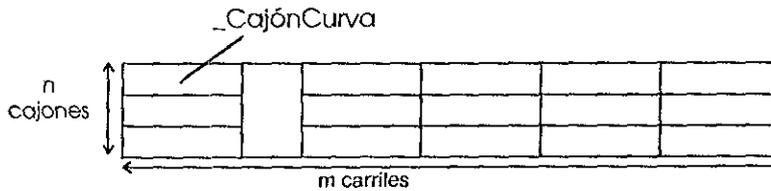


Fig. 5.10. Dimensiones del CURVÍGRAFO.

Para construir un `_CajónCurva` son necesarios los atributos de la Curva (✓).

```
struct _AtributosCurva
{
String Nombre; ✓
String Unidad;
double EscalaIzquierda; ✓
double EscalaDerecha; ✓
double CambioEscalaIzquierda;
double CambioEscalaDerecha;
int TipoEscala;
int TipoDeLinea; ✓
int ColorCurva; ✓
int MallaHorizontal;
int ColorMallaHorizontal;
int MallaVertical;
int ColorMallaVertical;
int CarrilInicial; ✓
int NumeroCarrilesOcupados; ✓
int Cajon; ✓
double IntervaloProfundidadInicial;
double IntervaloProfundidadFinal;
bool Visible; ✓
};
```

Un `_CajonCurva` (Fig. 5.11) es una estructura formada por los siguientes elementos:

```
struct _CajonCurva
{
    TBevel      *Marco
    TLabel      *Escalazq
    TLabel      *Nombre
    TLabel      *EscalaDer
    TLabel      *Color
    TLabel      *Sombra
};
```

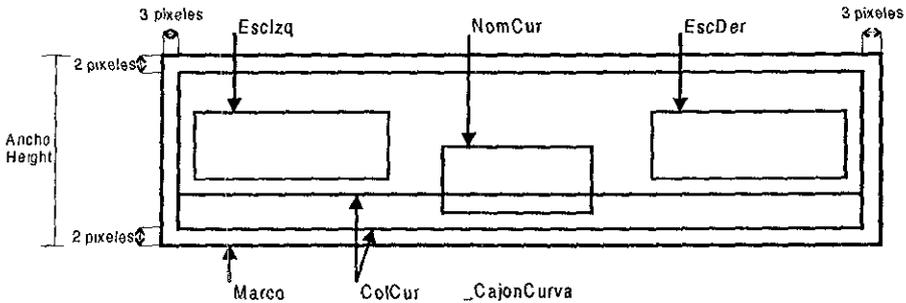


Fig. 5.11. Elementos que forman un `_CajonCurva`.

`Esczq`, `NomCur`, `Sombra`, `EscDer` y `ColCur` son derivados de la clase `TLabel`, esto debido a que esta clase presenta todas las propiedades y métodos para manejar y controlar letreros dentro de la ventana. `Marco` es derivado de la clase `TBevel` y determina el ancho del cajón.

Elemento	No. Pixeles Ancho (Height)	No. Pixeles Largo (Width)
<code>EsczqXY</code>	11	40
<code>EscDerXY</code>	11	40
<code>NomCurXY</code>	16	30
<code>ColCurXY</code>	20	variable
<code>MarcoXY</code>	28	variable

`XY` es determinado por la posición dentro del arreglo `_CajonCurva` que forman el CURVÍGRAFO.

A continuación se muestra la tabla donde se muestran las formulas para calcular cada uno de los atributos que forman el CURVIGRAFO.

Largo Sombra→width	Largo Marco width	Largo ColCur→width	Posición Esclzq→Left	Posición ColCur→Left	Posición NomCur→left	Posición EscDer→left
LargoDelCaml	LargoDelCaml	Marco→width-6	Marco→Left+2	Marco→Left+2	Marco→Left+ Marco→Width/2-15	Marco→Left+ Marco→Width/2-4

Para la organización de los cajones es necesario considerar la posición del carril de profundidad, ya que éste no tiene las mismas dimensiones de un _CajonCurva. Además también se debe consultar si una curva ocupa más de un carril, debido a que por medio del número de carriles ocupados (NumeroCarrilesOcupados) se determina el ancho del cajón. Por otra parte el carril de profundidad no puede estar contenido entre dos carriles ocupados por la misma curva. Fig. 5.12.

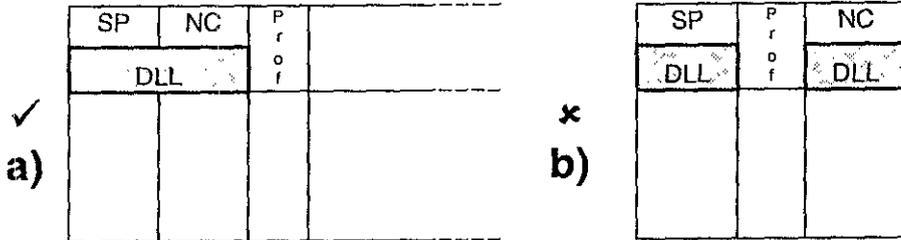


Fig. 5.12. Presentación de una curva que ocupa dos carriles. a)Correcta. B)Incorrecta

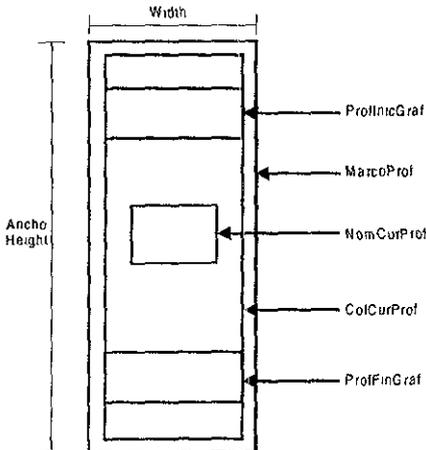


Fig. 5.13 Elementos de _CajonCurvaProfundidad

El ancho del cajón del carril de profundidad depende directamente del número de cajones (NumeroDeCajones) que contiene la gráfica (de 1 a 5) y de la posición del carril de profundidad. El largo de éste depende del largo de los cajones de las curvas, y su valor aproximado es la tercera parte del largo de un _CajonCurva (Fig. 5.13). Para determinar los atributos del carril de profundidad se consideran los valores de la tabla.

Elemento	No. Pixeles Ancho (Height)	No. Pixeles Largo (Width)	→Left	→Top
ProfIniGraf	11	40	$\text{MarcoProf} \rightarrow \text{left} + \frac{\text{MarcoProf} \rightarrow \text{Width} - 20}{2}$	-----
ProfFinGraf	11	40	$\text{MarcoProf} \rightarrow \text{left} + \frac{\text{MarcoProf} \rightarrow \text{Width} - 20}{2}$	$\text{MarcoProf} \rightarrow \text{Height} - 18$
NomCurProf	16	30	$\text{MarcoProf} \rightarrow \text{left} + \frac{\text{MarcoProf} \rightarrow \text{Width} - 15}{2}$	$\frac{\text{MarcoProf} \rightarrow \text{Height} - 8}{2}$
ColCurProf	$\text{AC} * \text{NumeroDeCajones} - 8$	$\frac{\text{Marco} \rightarrow \text{Width} - 6}{3}$	$\text{MarcoProf} \rightarrow \text{left} + 2$	-----
MarcoProf	$\text{AC} * \text{NumeroDeCajones}$	$\frac{\text{Marco} \rightarrow \text{Width}}{3}$	$\text{Marco} \rightarrow \text{Width} * (\text{PCP} - 1) - (\text{PCP} - 1)$	-----
SombraProf	$\text{AC} * \text{NumeroDeCajones}$	$\frac{\text{Marco} \rightarrow \text{Width}}{3}$	$\text{Marco} \rightarrow \text{Width} * (\text{PCP} - 1) - (\text{PCP} - 1)$	-----

PCP = Posición del carril de profundidad, AC = Ancho_Del_Cajon.

5.5 VISOR

El visor es la parte dentro de la cual se grafican las curvas generadas a partir de los datos de un archivo LAS, compuesto por carriles. Una curva puede ser graficada dentro de uno o más carriles. El visor y el curvígrafo deben estar sincronizados, debido a que si una curva es movida de un carril a otro, automáticamente el curvígrafo modifica la posición del cajón de la curva, como se muestra en la figura 5.14.

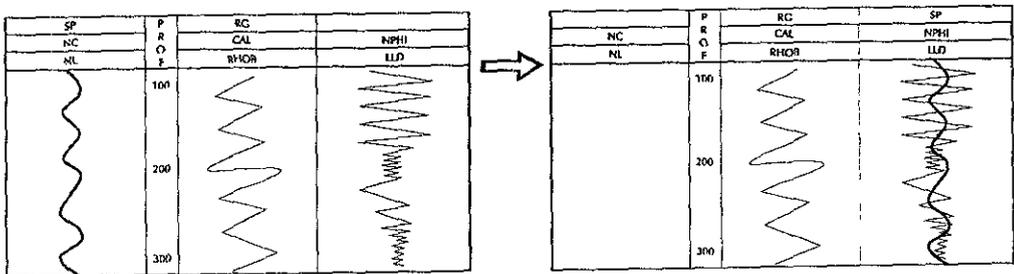


Figura 5.14. Cambio de carril de una curva (SP).

Es conveniente que CURVÍGRAFO y VISOR estén contenidos en la misma ventana, para ello es necesario la inclusión de dos "scrollbar", una horizontal y otra vertical que controlen la información de la gráfica en ambas direcciones, como se muestra en la figura 5.15.

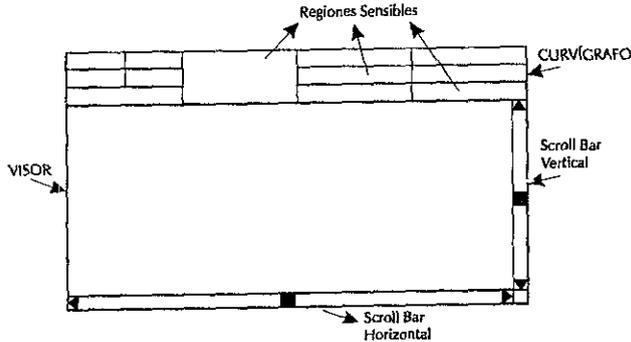


Figura 5.15. Presentación del VISOR y Del CURVIGRAFO en la misma ventana

El scroll bar vertical desplazará la gráfica (VISOR) en el eje "y" y las regiones sensibles (CURVÍGRAFO) no sufrirán cambios debido a que deben permanecer visibles siempre al usuario para conocer que parte de la gráfica esta observando.

El scroll bar horizontal desplazará la gráfica (VISOR) en el eje "x", por lo que las regiones sensibles (CURVÍGRAFO) se desplazarán, debido a que se encuentran coordinadas con la información de la gráfica presentada en el VISOR.

Los dos sroll bars deben ser redimensionables y reacomodables en tiempo de ejecución, esto debido a que el número de cajones es variable y además se deben adaptar al tamaño de la ventana.

Para manejar los datos de cada una de las curvas contenidas en el archivo LAS o en el metafile se generan polilíneas con los siguientes elementos:

Atributos	Descripción
IL	Identificador de la línea
Color	Color con el que se pinta la línea
Style	Estilo ó tipo con el que se dibuja la línea
Width	Anchura en pixeles de la línea

IL ColorLinea EstiloLinea AnchuraLinea (-,-) (-,-) (-,-) (-,-) (-,-)

lista de vectores que forman una polilínea

Por lo tanto una polilínea se define como:

```

struct POLILINEA
{
int lL; //Identificador de la línea
TColor ColorLinea;
TPenStyle EstiloLinea;
int AnchuraLinea;
struct_Vector *VectorInicial;
struct_Vector *VectorFinal;
}

struct _PuntoXY
{
float X;
float Y;
};

struct _Vector
{
int
struct _PuntoXY *coordenada;
struct_Vector *Sig;
};
    
```

Lista ligada de vectores que forman la polilínea

A partir de los puntos de la gráfica, se elabora una lista ligada, dentro de la cual cada elemento contiene la dirección del siguiente punto. El final de la línea se determina cuando la dirección del siguiente punto es nula (Fig. 5.16).

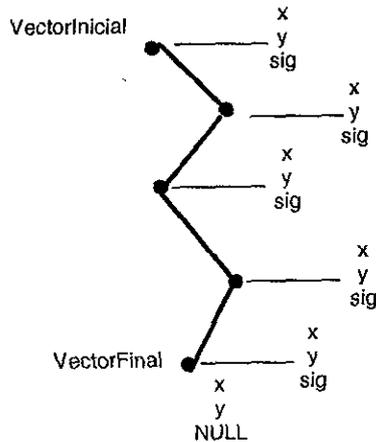
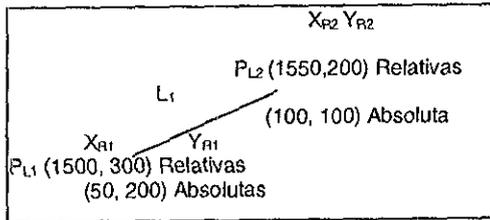


Fig. 5.16. Polilínea formada por una lista ligada de vectores.

Para la presentación de la información se manejan coordenadas relativas, tal y como lo muestra la fig. 5.17.



$P_{L1} (X_{R1} - X_{Relativa}, Y_{R1} - Y_{Relativa})$

$P_{L2} (X_{R2} - X_{Relativa}, Y_{R2} - Y_{Relativa})$

$L_1 (P_{L1}, P_{L2})$

Fig. 5.17. Manejo de coordenadas relativas para la presentación de las polilíneas.

Al mover los *scroll bar* horizontal y vertical, lo único que se modifican son las coordenadas relativas ($X_{Relativa}$, $Y_{Relativa}$)

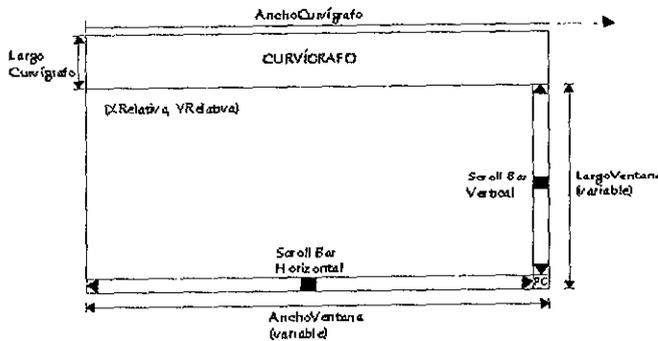


Fig. 5.18. Elementos empleados para el manejo de las coordenadas relativas.

Calculo de las dimensiones y posiciones de los elementos que controlan el VISOR y el CURVÍGRAFO. Fig. 5.18.

AnchoGráfica = AnchoCurvógrafo

LargoGráfica = numero de datos * escalaTotalGrafica .

LargoCurvógrafo = NumeroDeCajones*AnchoDelCajon.

Para delimitar los rangos de los *Scroll Bars*.

Scroll Bar Horizontal -> Min = 0

Scroll Bar Horizontal -> Max = AnchoGráfica - AnchoVentana + 20 (Ancho del ScrollBar)

Scroll Bar Vertical -> Min = 0

Scroll Bar Vertical -> Varía según la bandera Curvógrafo Visible

if (CurvógrafoVisible)

 ScrollBarVertical -> Max = LargoGráfica - LargoVentana + LargoCurvógrafo

else

 ScrollBarVertical -> Max = LargoGráfica - LargoVentana

Posición del *thumb* de los *scroll bars* y control de las coordenadas relativa.

HORIZONTAL

El Desplazamiento del CURVÍGRAFO y de la gráfica horizontalmente se controla con el *scroll bar* Horizontal, este desplazamiento está almacenado en:

Scroll Bar Horizontal -> Position

Por medio del movimiento del scroll Bar Horizontal también se actualiza la coordenada relativa "X" (XRelativa)

XRelativa = Scroll Bar Horizontal -> Position

Para el Desplazamiento Horizontal no importa si el CURVÍGRAFO está visible o no, simplemente se desplaza la gráfica y el CURVÍGRAFO el número de unidades correspondientes.

VERTICAL

El desplazamiento vertical es únicamente de la gráfica, esto debido a que el CURVÍGRAFO siempre debe permanecer "visible" (en la misma posición).

El desplazamiento de la gráfica verticalmente está almacenado en:

Scroll Bar Vertical -> Position

Por medio de este desplazamiento se actualiza la coordenada relativa "Y" (YRelativa).

Para determinar el valor de la YRelativa, únicamente se toma en cuenta el valor de la posición del *thumb* del scroll bar vertical.

YRelativa = ScrollBarVertical → Position

Cada una de las gráficas debe considerar dos ventanas: una ventana con las características generales (CARACTERIZADOR) y la otra con la gráfica (CURVÍGRAFO Y VISOR); como el sistema debe soportar el manejo de varias gráficas simultáneamente, implica la utilización de muchos recursos del sistema (memoria, acceso a disco, despliegue de la gráfica, etc.); por tal motivo se considera conveniente restringir el sistema a un número limitado, por ejemplo 4 y dependiendo de la funcionalidad del sistema incrementarlo posteriormente.

También se deben considerar los acercamientos/alejamientos que se realicen sobre una gráfica. Es recomendable realizar un solo acercamiento/alejamiento a la vez ya que varios pueden causar confusión al usuario. Una de las alternativas es deshabilitar las demás partes del sistema (ventanas), hasta que la ventana dentro de la cual se encuentra el acercamiento/alejamiento de la gráfica se ha "cerrado". Como se muestra la figura 5.19.

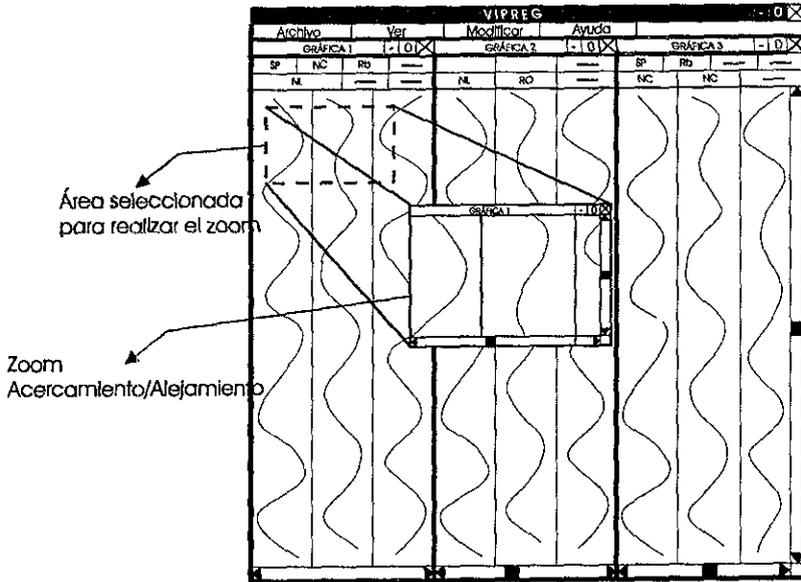


Fig. 5.19. Realización de un acercamiento/alejamiento de la gráfica.

El CURVÍGRAFO tiene la opción de ocultarse dentro de la ventana para sí maximizar el espacio visible de la gráfica, figura 5.20.

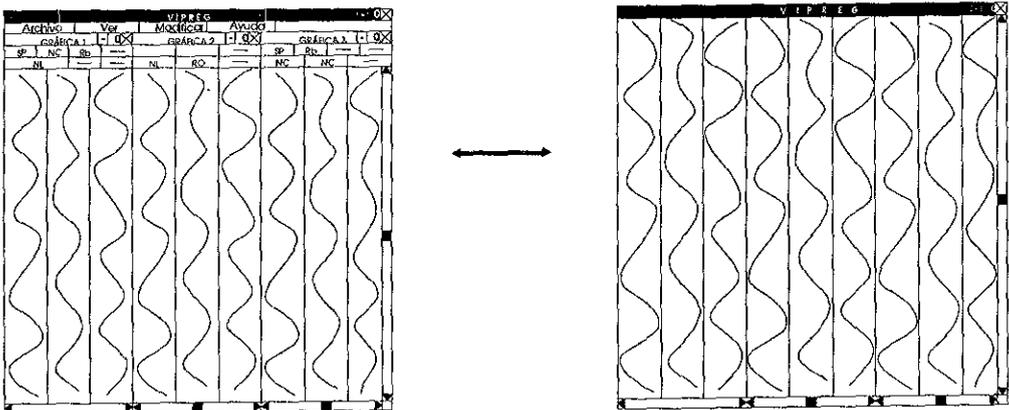


Fig. 5.20. Ocultamiento de los cajones de la gráfica.

El *scroll bar* Horizontal desplaza el CURVÍGRAFO (los cajones) hacia la izquierda y/o derecha para explorar la gráfica. El CURVÍGRAFO lee un arreglo de estructuras que contienen los atributos de las curvas para formar los cajones y presentar la información cuando se seleccione modificar los atributos de una curva en particular. A continuación se definen las funciones miembro de la clase CURVÍGRAFO.

Class CURVÍGRAFO : Public TForm

```
{
// Para controlar todos los cajones del CURVÍGRAFO y los atributos de la curva
struct _CajonCurva [Maximo_Numero_De_Cajones] [Maximo_Numero_De_Carriles];
// Para controlar los atributos de la curva de profundidad
siruct_CajonCurva CajonProfundidad;
int LargoDelCarril; // Para almacenar el largo de cada carril
TCARACTERIZADOR *PropGenerales; //Para presentar las características generales.
```

Método	Actividad
presentaPropiedadesGenerales (void)	Presenta el diálogo con las propiedades generales de la gráfica. En este diálogo no se pueden modificar las propiedades generales de la gráfica.
inicializaCajones(void)	Inicializa el arreglo Cajon [] [] para controlar todos los cajones.
inicializaScrollBars (void)	Inicializa los scroll bars Horizontal y Vertical de acuerdo al tamaño de la ventana y las dimensiones de la gráfica (Número de carriles y de Cajones).
mueveCajones (int incrementa)	Desplaza todos los cajones según el valor de incremento + a la derecha y - a la izquierda.
ocultaTodosLosCajones (void)	Oculto todos los cajones para maximizar el espacio de despliegue de la gráfica.
visualizaCajonesdeLaGrafica (void)	Visualiza los cajones de la gráfica según el número de carriles y cajones que posea en sus propiedades generales.
redimensionaCajones (int largoCajon)	Cambia el largo de todos los cajones de la gráfica. Reorganiza los cajones de acuerdo a la posición del carril de profundidad.
PresentaDialogoModificarPropiedades Generales (void)	Presenta un diálogo con las propiedades generales de la gráfica, dentro del cual se pueden hacer modificaciones. Se ejecutan las funciones para realizar las modificaciones seleccionadas.
PresentaDialogoModificarAtributosDeLasCurvas (avoid)	Presenta un diálogo con los atributos de las curvas dentro del cual se pueden hacer modificaciones. Se ejecutan las acciones para realizar las modificaciones seleccionadas.
Constructor TCURVÍGRAFO (TComponent *, String NombreDeArchivoLAS).	

Crea una ventana de acuerdo a los datos especificados en el archivo LAS que se le pasa como argumento, determinando las estructuras _DatosGrafica y _AtributosCurva. Considerando Numero de Carriles, Numero de Cajones, Posición del Carril de Profundidad, colocando los atributos de las curvas en su carril y cajón correspondiente.

scrollBarHorizontalCambio ()	Función que se ejecuta cuando el Scroll Bar Horizontal ha sufrido algún evento. Por medio de esta función se recorren los cajones y la gráfica horizontalmente.
cambioTamañoVentana ()	Se ejecuta cuando la ventana cambia de tamaño, por lo que los scroll bar Horizontal y Vertical se deben ajustar al tamaño actual de la ventana.
modificaAtributosCurvas ()	De acuerdo al cajón, se cargan los atributos de las curvas contenidos en éste y se presentan en el diálogo modificar atributos de las curvas.
BotonPropiedadesGenerales ()	Presenta las propiedades generales de la gráfica en un diálogo del tipo TCARACTERIZADOR, en este diálogo no se pueden modificar las propiedades generales.
repintanForma ()	Función que se ejecuta cada que existe un refrescamiento de la información de la ventana.

5.6 DIÁLOGO PARA MODIFICAR LAS PROPIEDADES DE LA GRÁFICA

The dialog box, titled "Modificar Prop. Generales.", contains the following elements:

- Fields for text input: Pozo, Campaña, Campo, Localidad, Provincia, Servicio, Fecha, Prof. Inicial, Prof. Final.
- Fields for dropdown selection: Escala, #Carriles, #Cajones, Colocación carril prof.
- Buttons at the bottom: "Aceptar" and "Cancelar".

Fig. 5.21. Diálogo para modificar las propiedades generales de la gráfica.

Este diálogo es empleado para modificar las propiedades generales de la gráfica (Fig. 5.21), como son: El nombre del pozo, la compañía, la escala general de la gráfica, el número de carriles, el número de cajones, la posición del carril de profundidad, etc.

La forma en que se lleva a cabo el proceso para modificar las propiedades generales es el siguiente:

1. recibe una estructura `_DatosGráfica` con las propiedades generales de la gráfica, carga los datos en el diálogo.
2. Se inicializa una estructura del tipo `_DatosGráfica` a nulo, en esta estructura se van almacenando los cambios que realiza el usuario.
3. Presenta el diálogo y el usuario modifica las propiedades generales.
4. Aquí existen dos posibilidades
 - Si selecciona *Cancelar*, no se efectúa ninguna acción y se cierra el diálogo, no se realiza ninguna actualización a las modificaciones hechas dentro del diálogo.

- Se selecciona *Aceptar* se retornan los cambios efectuados para que se realicen las acciones correspondientes, se actualizan las propiedades generales de la gráfica y se cierra el diálogo. Los cambios se retornan en la estructura *_DatosGráfica* (*DetectaCambios*), en la cual los valores que no cambiaron con respecto a la estructura que se paso como argumento tienen valor "NULL" o "\0" según el tipo, la función que realiza el llamado correspondiente determina las acciones o procesos que se realizan con cada uno de los diferentes cambios efectuados. La definición de la clase y los métodos que emplea se muestran a continuación.

Class PROPIEDADES_GENERALES Public TForm

```
private: // User declarations
struct _DatosGráfica DetectaCambios;
struct _DatosGráfica AlmacenaDatosEntrada;
bool banderaCambios;
bool banderaCambioProfInicial;
bool banderaCambioProfFinal;
bool verificaCambios(void);
```

Métodos que se emplean:

TPROPIEDADES_GENERALES(TComponent* Owner);	Es el constructor y crea un diálogo con textos y combobox donde pueden ser modificadas las propiedades generales.
TPROPIEDADES_GENERALES(TComponent* Owner, struct _DatosGráfica PropiedadesGeneralesGráfica);	Constructor, crea el diálogo y recibe como argumento una estructura con las propiedades generales, cargandolas y presentandolas dentro del diálogo.
void cargarPropiedadesGenerales(struct _DatosGráfica Propiedades);	Carga los valores de las propiedades generales contenidas en una estructura <i>_DatosGráfica</i> en el diálogo.
struct _DatosGráfica obtenPropiedadesGenerales (struct _DatosGráfica Propiedades);	Recibe como argumento las propiedades generales cargandolas y presentandolas dentro de un diálogo para que el usuario realice las modificaciones y retorna una estructura con las modificaciones.
void presentaPropiedadesGenerales(struct _DatosGráfica Propiedades);	Únicamente presenta las propiedades generales en el diálogo y los cambios efectuados no tienen efecto. Sirve para comprobar que los cambios efectuados se realizan y almacenan correctamente.
void inicializaEstructuraCambios(void);	Inicializa la estructura <i>DetectaCambios</i> (<i>_DatosGráfica</i>) a nulos, para almacenar los cambios.

5.7 DIÁLOGO PARA MODIFICAR LOS ATRIBUTOS DE LAS CURVAS

ATRIBUTOS DE LA CURVA	
Nombre	<input type="text"/>
Tipo Escala	<input type="text"/>
Escala Derecha	<input type="text"/>
Escala Izquierda	<input type="text"/>
Color	<input type="text"/> <input checked="" type="checkbox"/>
Malla Horizontal	<input type="text"/> <input checked="" type="checkbox"/>
Color Malla Hor.	<input type="text"/> <input checked="" type="checkbox"/>
Malla Vertical	<input type="text"/> <input checked="" type="checkbox"/>
Color Malla Vert.	<input type="text"/> <input checked="" type="checkbox"/>
Carril Inicial	<input type="text"/> <input checked="" type="checkbox"/>
# Carriles Ocupados	<input type="text"/> <input checked="" type="checkbox"/>
Rellenos	<input type="text"/> <input checked="" type="checkbox"/>
Intervalo	<input type="text"/> <input checked="" type="checkbox"/>
Prof. Inicial	<input type="text"/>
Prof. Final	<input type="text"/>
Visible	<input type="text"/> <input checked="" type="checkbox"/>

Anterior Curva Siguiete Curva

Aceptar Cancelar Default

Fig. 5.22. Diálogo para modificar los atributos de las curvas.

Este diálogo sirve para modificar los atributos de las curvas (Fig. 5.22) como son: El nombre de la curva, el tipo de línea, el color de la línea, la escala izquierda, la escala derecha, el tipo de curva, el cajón y el carril que ocupa, etc.

Su funcionamiento es similar al diálogo para modificar las propiedades generales de la gráfica y sigue la siguiente secuencia.

1. Recibe como argumento un arreglo de estructuras del tipo AtributosCurva, el cual carga dentro del diálogo.
2. Inicializa un arreglo de estructuras del tipo AtributosCurva a nulo, el cual sirve para ir almacenando los cambios.
3. El usuario efectúa las modificaciones realizando el recorrido del arreglo de los atributos de las curvas a través de los botones Anterior Curva y Siguiete Curva.
4. Pueden ocurrir tres casos:
 - Si selecciona Cancelar se retorna y las modificaciones realizadas con respecto a los datos de entrada no tienen efecto.
 - Si selecciona Aceptar se retorna la estructura que contiene los cambios efectuados, en los atributos que no existió cambio se retorna un valor nulo. La función que llamó a este componente es el encargado de efectuar las acciones y actualizaciones correspondiente por cada una de las modificaciones realizadas por el usuario.

- Si selecciona Default se cargan los valores por default de la curva, los cuales se encuentran en un archivo previamente almacenado con los valores mas comunes de cada una de las curvas que son mas comúnmente utilizadas en el proceso de registros geofísicos.

Class ATRIBUTOS_CURVA

```
struct _AtributosCurva ArregloCurvas[20];
struct _AtributosCurva DetectaCambiosCurvas[20];
int IndiceActual;
```

TATRIBUTOS_CURVAS(TComponent* Owner);	Constructor. Crea el diálogo para modificar los atributos de las curvas.
struct _AtributosCurva obtenAtributosCurva(struct _AtributosCurva curva[20], int indice);	Recibe como argumento un arreglo de estructuras con los atributos de las curvas, lo presenta dentro del diálogo y retorna los cambios efectuados en una estructura.
void inicializaEstructuraDetectaCambios(void);	Inicializa un arreglo de estructuras del tipo _AtributosCurva a nulo, cuando se efectua algun cambio se almacena en esta estructura.
void cargarAtributosCurva(struct _AtributosCurva curva);	Carga los atributos de una curva dentro del diálogo.
void presentaAtributosCurva(struct _AtributosCurva curvas[20], int indice);	Presenta los atributos de la curva con en indice que se pasa como argumento, se emplea para ir recorriendo el arreglo de los atributos de las curvas por medio de los botones Curva Anterior y Curva Siguiente.

6. PRUEBAS

Este capítulo presenta los resultados obtenidos en las pruebas para la validación y verificación del browser construido en el capítulo 5. El equipo en el cual se realizaron las pruebas tiene las siguientes características: Una PC con procesador pentium, 75 MHz, 16 Mb en RAM, 2.0 Gb en disco duro (espacio libre de 400 Mb), monitor a color SVGA (14") con Sistema operativo Windows 95.

Para la graficación de los archivos LAS se tenía como primera opción la "segmentación" de la información, pero al realizar pruebas dentro de la programación, se consideraron varias posibilidades, entre ellas esta la creación de polilíneas formadas de listas simplemente ligadas, se comenzó por probar 500 elementos generados aleatoriamente, se continuo incrementando el número de nodos y de polilíneas, hasta generar 14 polilíneas con 20,000 nodos c/u y el sistema se comporto satisfactoriamente, cabe destacar que si no se hubiese tenido estas consideraciones tal vez aumentaría la complejidad y tiempo de desarrollo del sistema.

Para tener una idea clara de la cantidad de información que se maneja, nos basamos en un archivo de 4,000 mts, con 12 curvas y un intervalo de muestreo de .3048 mts. En este caso estamos hablando de dos valores (X,Y) del tipo flotante (32 bits), por lo que necesitamos la siguiente memoria para almacenar esta información:

$$\frac{4000m*(12curvas*2puntosXY*4bytes)}{.3048 m} = 1,259,850 \text{ bytes por gráfica aprox.}$$

Se hicieron pruebas de sobrecarga para verificar la cantidad de gráficas que se pueden tener abiertas al mismo tiempo, lográndose obtener hasta 10 gráficas con las características descritas anteriormente y el sistema se comporto satisfactoriamente. También se comprobó la utilización del sistema con algunas otras aplicaciones abiertas con resultados satisfactorios.

En relación a la validación de datos se realizaron las siguientes pruebas: especificación de las escalas de las curvas, profundidades inicial y final gráficas, datos correctos, etc.

Para la verificación de las modificaciones realizadas a las características generales de la gráfica y los atributos de las curvas, el sistema contempla un estado de autopruueba y presenta las modificaciones realizadas al colocar la bandera autopruueba a "TRUE.

Debido a restricciones al acceso de información de Registros de Pozo almacenados en formato LAS, sólo fue posible comprobar la funcionalidad del sistema con un volumen de 11 archivos, aunque es una muestra pequeña, puede afirmarse que el sistema se comporto satisfactoriamente, los resultados de las pruebas se presenta en la tabla de la Fig. 6.1.

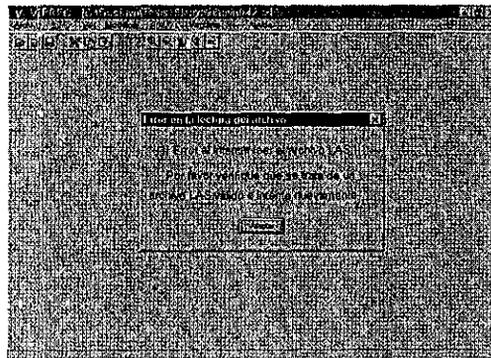
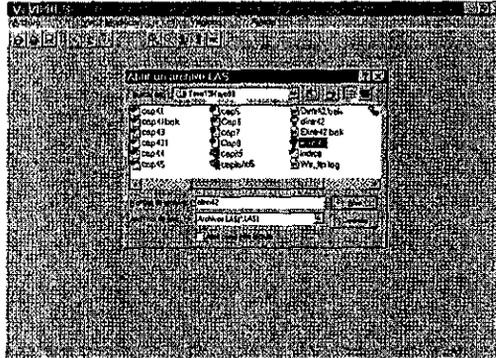
Nombre del archivo LAS	cargó correctamente el encabezado del archivo Prop. Grs.	cargó correctamente los nombres y los atributos de las curvas	cargó correctamente los datos de las curvas	presentó el caracterizador con los datos conexos	desplegó la gráfica (visor y curvgrafo) adecuadamente	realizó modificaciones y actualizaciones adecuadamente
esc1	✓	✓	✓	✓	✓	✓
mer1	✓	✓	✓	✓	✓	✓
mer11	✓	✓	✓	✓	✓	✓
mer1d	✓	✓	✓	✓	✓	✓
mer21	✓	✓	✓	✓	✓	✓
mer31a	✓	✓	✓	✓	✓	✓
mer41	✓	✓	✓	✓	✓	✓
mer51	✓	✓	✓	✓	✓	✓
mer71	✓	✓	✓	✓	✓	✓
mer81	✓	✓	✓	✓	✓	✓
topo61	✓	✓	✓	✓	✓	✓

Fig. 6.1. Resultados de las pruebas realizadas a un conjunto de archivos en formato LAS.

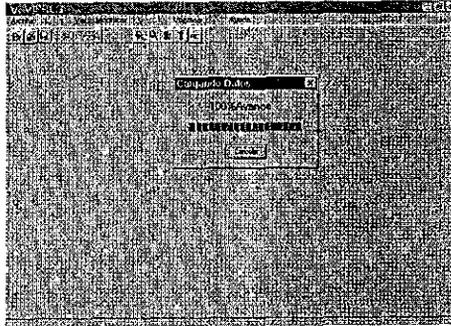
El VIPREG cuenta con un analizador dentro del proceso de creación de la gráfica que permite reconocer si el archivo de entrada se encuentra en formato LAS, realizando las siguientes acciones:

- Si es un archivo en formato LAS, realiza la serie de pasos para elaborar y desplegar la gráfica correspondiente.
- Si no es un archivo en formato LAS, el sistema cancela inmediatamente todos los procesos y no se genera la gráfica, presentando un diálogo con la información correspondiente del error ocurrido (Fig. 6.2).

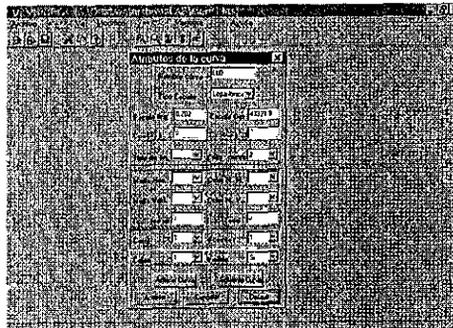
Se intentó graficar archivos que no pertenecen al formato LAS y el sistema respondió satisfactoriamente no elaborando la gráfica sin sufrir alguna alteración o mal funcionamiento, por lo que se puede afirmar que la validación del formato del archivo de entrada permite seguridad y confiabilidad.



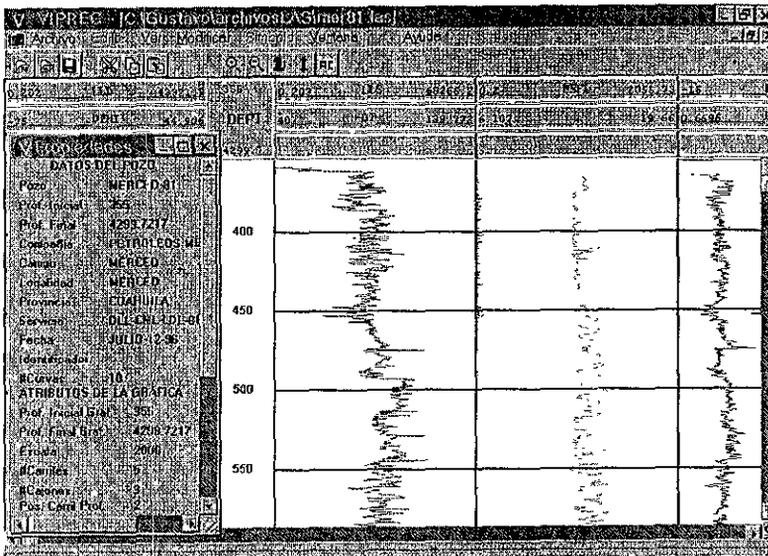
Fi. 6.2. Mensaje de error en el formato del archivo de entrada.



Paso 4. Lectura de los datos de las curvas.



Paso 5. Presentación de los atributos de las curvas.



Paso 6. Presentación de la gráfica con las características seleccionadas.

7. CONCLUSIONES

Durante el presente trabajo se lograron los objetivos planteados en un principio, adquiriendo gran experiencia tanto en el área de computación como en ingeniería geofísica y petrolera, conociendo a grandes rasgos las características principales de estas áreas tan importantes dentro del país.

El objetivo primordial se cumplió con gran éxito, lo comprueban los resultados obtenidos y presentados con la construcción del "modelo básico" del browser gráfico para la visualización de la información petrolera (VIPREG), el cual trabaja eficientemente y permite realizar todos los requerimientos planteados. Se pueden comparar estos resultados con dos versiones que se manejan dentro del IMP (Instituto Mexicano del Petróleo).

El rtotal97 elaborado en un lenguaje C y C++ que trabaja sobre una workstation en una plataforma UNIX, que aunque ya maneja el contexto de vectores no está profundamente vinculado con la modificación en tiempo real de las características de la gráfica, esto debido a que para modificarlas se requiere de la reconstrucción total de la misma, lo que ocasiona un gran número de procesos muy elaborado para la descomposición y reorganización de la "nueva gráfica", produciendo una gran utilización del tiempo de proceso que en algunos casos resulta molesto al usuario.

El WIMPLog es una adaptación de una versión pasada del rTotal97 para trabajar en una PC con sistema operativo Windows 95, este sistema procesa archivos LAS, pero es un sistema muy restringido para el manejo de las características de la gráfica, debido a que rasteriza horizontalmente toda la gráfica presentándola únicamente como una imagen estática a la cual no puede hacerse ninguna adaptación en tiempo real. Como la gráfica es presentada rasterizada, la información de cada uno de los elementos (curvas), que forman la gráfica se pierden y es imposible distinguir entre un elemento u otro.

Existen también algunos sistemas comerciales que grafican archivos LAS, de los cuales se han probado algunos demos en las instalaciones del IMP y no cumplen con todas las características que se requieren para interpretar y manejar adecuadamente las gráficas, o en algunos casos son muy complejos en su utilización y al usuario se le dificulta en gran medida el empleo de estos.

El VIPREG (Visualizador de Información Petrolera: Registros), ofrece las siguientes ventajas con respecto a los sistemas antes descritos:

- Es una herramienta de apoyo para la graficación de archivos LAS (formato en el cual se almacenan los registros geofísicos de pozo),
- Flexibilidad en la adecuación de la gráfica a las necesidades del usuario, empleando el formato en que hasta la fecha se grafican los registros
- Presenta un ambiente más agradable al usuario por medio de ventanas que permite realizar las modificaciones en tiempo real con un mínimo de tiempo de proceso .
- Ésta desarrollado bajo una metodología orientada a objetos que en un futuro facilitará la adecuación, actualización y reutilización de sus componentes en una forma mas transparente y sencilla que ahorrará un gasto considerable en tiempo, dinero y esfuerzo.
- El tiempo de despliegue de la gráfica es mínimo.
- Permite guardar la gráfica procesada (metafile) que puede ser presentada posteriormente por el sistema y que conserva todas las características propias que se le asignaron cuando se creo.
- Trabaja en una computadora personal con sistema operativo Windows95, equipo que es mas comúnmente utilizado en cualquier instalación de PEMEX, a diferencia de las estaciones de trabajo que son escasas y su adquisición mucho mas costosa.

Uno de los objetivos paralelos, aunque no el más importante fue comprobar la funcionalidad de una nueva metodología orientada a objetos titulada "Desarrollo de Software Orientado a Objetos con Calidad" (DSOOC), al final del presente trabajo se puede concluir que funciona bien, el compararla con otras metodologías resulta confuso e inapropiado porque no es tema de discusión de este trabajo la eficiencia de las metodologías, pero lo que si se puede afirmar con claridad es que la metodología permite tener un control sobre el análisis, diseño e implementación del sistema elaborando paralelamente a estos procesos el manual del sistema.

8. BIBLIOGRAFÍA.

1. IEEE Software
Marzo de 1995 Pags. 21-32
Image - Browser Taxonomy and
Guidelines for Designers
Catherine Plaisant, Davis Carr, and
Ben Shneiderman
University of Maryland.

2. Principios/Aplicaciones de la
Interpretación de Registros.
Shlumberger.

3. A sonic method for analyzing the quality
of cementation of borehole casings
Gros mangin, M.; Kokesh, F.P.; Majani, P.
Petroleum Transactions AIME, 1961.

4. Interactive Specification of Flexible
User Interface Displays.
S. Hudson and S. Mohamed.
ACM Trans. Information Systems, July 1990.
Pgs. 269-288.

5. C++ Builder
Borland
Version 1.0

6. Borland C++
User's Guide
Borland
Version 5.
Volumen 1

7. Microsoft. Visual C++
Programmer's Guide.
Developmet Systems for Windows

8. Calidad de Software, el enfoque
con tecnología orientada a objetos.
Alfonso Miguel Reyes
UIA 1998.

9. Software Engineering Object Oriented
ACM
Jacobson.-