

Desarrollo de un Visualizador de Señales de Radar

Acosta, Nelson; Tosini, Marcelo y Mezzanotte, Martín

Instituto INTIA – Departamento de Computación y Sistemas

- Facultad de Ciencias Exactas -

Universidad Nacional del Centro de la Provincia de Buenos Aires

Tandil – CP 7000 - Buenos Aires - Argentina

E-mail: {nacosta, mtosini, mmezzano}@exa.unicen.edu.ar

Resumen

El presente trabajo consiste en el desarrollo de un sistema de visualización de información para interfaces de sistemas de radar. El mismo es un sistema de software flexible y portable, que permite ser empleado en radares de diferentes tecnologías y capaz de adaptarse eficientemente a las necesidades específicas de cada dominio de aplicación.

El sistema permite también implementar algoritmos de procesamiento de información y reconocimiento automático de objetivos.

Palabras clave: Radar, Visualización de Información, Procesamiento de Señales.

1 INTRODUCCIÓN

Conceptualmente, un sistema de radar consta de cinco componentes: un generador, un receptor, un amplificador, un analizador o procesador y un visualizador [1]. En el presente trabajo se describe el desarrollo de un sistema de software cuyo objetivo es reemplazar los últimos dos componentes; de forma tal de poder actualizar y modernizar las prestaciones actuales de diferentes radares, como por ejemplo: radares de control de tráfico aéreo, radares tácticos de uso militar o civil, radares de buques, entre otros.

Las funciones del analizador de un sistema de radar consisten en obtener de las señales recibidas la información deseada, determinando si las reflexiones obtenidas por la antena se corresponden con objetos de interés para el sistema. Los componentes de análisis de los sistemas modernos llevan a cabo una gran variedad de funciones que permiten sintetizar la información deseada de manera simple y eficiente; como ser eliminación de ruido, predicción de trayectorias de objetivos, identificación de objetos y demás.

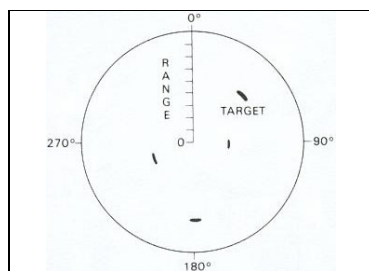


Figura 1-A: Presentación Plan Position Indicator

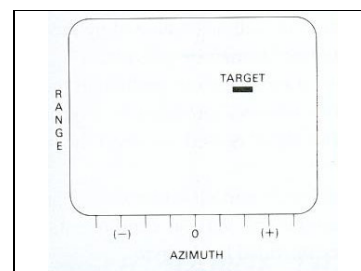


Figura 1-B: Presentación Tipo "B"

El visualizador de un radar tiene por objetivo presentar en una pantalla o display la información procesada por el analizador. Esta presentación puede realizarse de diferentes formas, según sean las necesidades del sistema. Por ejemplo, los radares de búsqueda o vigilancia que efectúan coberturas de 360° suelen utilizar presentaciones de tipo *Plan Position Indicator* (PPI) (Véase Figura 1-A), las cuales muestran los objetivos en forma polar centrado en la posición del radar; los radares tácticos de funcionamiento Aire-Aire, en cambio, suelen utilizar presentaciones de tipo "Bote" o "B" (Véase Figura 1-B), las cuales ubican en el display los distintos objetivos según la información de distancia (línea vertical) y azimuth (línea horizontal). [1]

En las secciones subsiguientes se describe detalladamente el trabajo de diseño y desarrollo de un sistema de visualización para señales de radares. Posteriormente, se detalla el trabajo de implementación del sistema, a modo de caso de estudio, adaptado a una plataforma tecnológica específica.

2 DISEÑO ARQUITECTÓNICO

El diseño arquitectónico del sistema fue desarrollado en base a un análisis detallado de las características funcionales y no funcionales requeridas.

2.1 Características Funcionales del Sistema

El sistema debe poder procesar señales provenientes de distintos dispositivos, así como también de otros componentes de software. Las señales a procesar por el sistema pueden ser clasificadas en los siguientes tipos:

- Señales analógicas y/o digitales provenientes del radar.

- Señales analógicas y/o digitales provenientes de dispositivos de hardware externos al radar.
- Señales digitales provenientes de componentes de software independientes, que puedan comunicarse con el sistema para propósitos específicos.
- Señales provenientes de los controles manuales del dispositivo soporte del sistema.

Las señales provenientes del dispositivo de radar son variadas y de características diversas. En primer lugar, se necesita conocer los valores de las reflexiones obtenidas por la antena del radar. Pero además, estos datos deben procesarse conjuntamente con la posición de la antena (lo cual depende del tipo de radar y de los modos de operación que el mismo provea) e información de sincronización de tiempos entre envío y recepción; de modo tal de poder componer una representación adecuada de la zona explorada por el radar. A esto se suman las señales provenientes de funcionalidades específicas provistas por el radar; como por ejemplo, las funciones de seguimiento (tracking) de objetivos, muy utilizadas en sistemas de radares para tráfico aéreo o radares tácticos de uso militar.

Frecuentemente, los sistemas de radar modernos se comunican con otros dispositivos para fines específicos. En radares meteorológicos o de tráfico aéreo es muy común –por ejemplo– complementar la información detectada por el radar con información geográfica o imágenes satelitales de la zona explorada. En radares instalados en buques o aviones resulta de vital importancia contar con información de navegación georeferencial, habitualmente provista por algún dispositivo o sistema externo (GPS, por ejemplo.). Por esto, es importante tener en cuenta al momento de diseñar y desarrollar un sistema de estas características que las señales de entrada pueden ser de naturaleza y procedencia sumamente heterogéneas.

Finalmente debe considerarse que los radares modernos cuentan con diferentes modos de operación, en los cuales puede variarse el funcionamiento mismo del dispositivo, alterando –por ejemplo– el movimiento de la antena y, por ende, la zona de exploración; o simplemente la información presentada en el display. Estos cambios en los parámetros de funcionamiento del sistema deben poder ser modificados por el operador del sistema, por medio de controles por ejemplo.

Como resultado del procesamiento de señales, el sistema debe presentar en un display información referente a los objetos detectados por el radar. Si bien la naturaleza de esta información depende del dominio de aplicación del sistema; es posible generalizar que el sistema debe presentar un conjunto de «objetos» y «propiedades», actualizando los cambios observados en los mismos y seleccionando aquellos atributos que puedan ser ignorados o destacados en la presentación.

En lo que respecta a las funcionalidades de interface, el sistema debe proveer una unidad de visualización multifunción (MFD – Multi Function Display) que pueda proyectar presentaciones gráficas flexibles y fácilmente modificables, que se adapten adecuadamente a las necesidades operativas de los usuarios. Para esto, resulta esencial poder definir diferentes tipos de presentaciones y configurar las características de las mismas según se desee.

2.2 Características No Funcionales del Sistema

Básicamente, este sistema debe llevar cabo un proceso de adquisición, interpretación y presentación de información en tiempo real. Para que el mismo pueda ser implementado con éxito, el diseño arquitectónico del sistema debe considerar una serie de condiciones que afectarán de manera crítica el desempeño del sistema.

En primer lugar, el sistema debe poder cumplir requisitos de performance sumamente estrictos. El sistema debe responder a las órdenes del usuario con la mayor celeridad posible y todos los cambios detectados en la información a presentar deben actualizarse en el MFD rápidamente. De modo tal de poder presentar en el display una representación correcta y precisa de la zona explorada.

En segundo término; el sistema debe poder ser actualizado con facilidad, en pos de adaptarse a diferentes cambios, tanto en lo que respecta al dominio de aplicación como a las necesidades de los

usuarios. Principalmente, el sistema debe adaptarse de manera sencilla para procesar señales provenientes de nuevos dispositivos de hardware o componentes de software, modificar los algoritmos de procesamiento de señales, cambiar o agregar formatos de presentación de información, entre otras funcionalidades importantes.

Otro aspecto central a tener en cuenta es la portabilidad del sistema. Es sumamente importante que el sistema pueda ser migrado a diferentes plataformas tecnológicas, adaptando su funcionalidad a distintos dominios de aplicación.

Finalmente, resulta importante considerar que el sistema debe ser implementado en una plataforma que provea un alto nivel de desempeño en la ejecución y sincronización de procesos en tiempo real.

2.3 Arquitectura de Software del Sistema

Las características buscadas en la arquitectura de software a desarrollar deben garantizar una fácil evolución del sistema a los posibles cambios del mismo a lo largo del tiempo, minimizar el impacto de estos cambios en los diferentes componentes y/o dispositivos que interactúan con el sistema e independizar las funcionalidades de procesamiento y visualización de la plataforma tecnológica subyacente. [2]

Tendiendo en cuenta estos objetivos y en función del análisis de las características funcionales y no funcionales descrito anteriormente, se diseñó una arquitectura de software basada en tres componentes: El primero es un componente de **adquisición** de información, que permite controlar diferentes dispositivos de hardware y recolectar señales analógicas y digitales provenientes de los mismos. El segundo es un componente de **procesamiento** desarrollado con diferentes algoritmos de interpretación, que permite identificar información relevante y seleccionar los elementos que deben ser visualizados. El tercero es un componente de **visualización**, que permite graficar información bajo diferentes configuraciones.

2.4 Componente de Adquisición

El Componente de Adquisición es responsable de implementar las funciones de lectura de las diferentes señales que debe procesar el sistema. Se encuentra compuesto –principalmente- de drivers de control de las distintas placas y dispositivos de hardware que brindan señales de entrada al sistema.

En este punto es importante considerar que las señales que deben ser procesadas por el sistema provienen de diferentes dispositivos (Por ejemplo: antena del radar, dispositivos de hardware asociados al radar, controles manuales, etc.), por lo cual el sistema debe poder sincronizar la recepción de señales disímiles y efectuar sobre los datos adquiridos las transformaciones necesarias para su futuro procesamiento.

2.5 Componente de Procesamiento

El componente de procesamiento tiene por objetivo procesar y analizar un conjunto de señales de diferentes características. Además, según las características del análisis que deba realizarse, algunas señales pueden requerir almacenamiento de información histórica (Algoritmos de seguimiento y predicción de trayectorias, estudio de frecuencia para reconocimiento de blancos, etc.). Por ende, este componente recibe por entrada un conjunto variable de datos que representan los distintos cambios en las señales de entrada y, procesamiento mediante, produce como resultado el conjunto de datos que representan los distintos cambios en la información a visualizarse en el MFD.

Las funciones de procesamiento que el sistema debe implementar dependen directamente del dominio de aplicación del mismo. Existen al respecto numerosas técnicas de procesamiento de señales que tienen aplicación directa en el desarrollo de sistemas de radares en distintos dominios.

Las técnicas basadas en análisis de Fourier, por ejemplo, tienen importantes aplicaciones en numerosos sistemas de radar y se han desarrollado al efecto diferentes métodos de cálculo computacional aplicables; como ser el método de Cooley-Tukey, ampliamente utilizado en para análisis de espectros y procesamiento digital de señales de radar. [1] En particular, las técnicas basadas en la Transformada Rápida de Fourier (FFT – *Fast Fourier Transform*) poseen variadas aplicaciones en este ámbito, siendo utilizadas en radares de pulso Doppler, radares de seguimiento de blancos móviles, en análisis de vibraciones en sistemas de radares láser, etc. [4]

En el caso de radares tácticos resulta de suma importancia el problema del seguimiento de objetivos, en sus diferentes variantes. Al respecto se han propuesto diferentes enfoques para resolver el problema de seguimiento de un único objetivo, de múltiples objetivos o clasificación de “falsas alarmas”, es decir recepciones que aparentan ser un objetivo y no lo son. Estos enfoques incluyen técnicas probabilísticas, estimación filtros de Kalman, entre otras. [3]

2.6 Componente de Visualización

El Componente de Visualización es responsable de implementar todas las funcionalidades de la Unidad de Visualización Multifunción (MFD). Por lo tanto, este componente no realiza ningún tipo de procesamiento o transformación de la información recibida, sino que provee funciones de configuración de presentaciones gráficas y modela en pantalla todo tipo de información que se estipule necesaria para el operador del sistema.

En este punto resulta importante señalar que existen numerosos tipos de presentaciones, los cuales son utilizados según las necesidades planteadas por los diferentes dominios de aplicación (Véase sección 1). Además, existen convenciones asociadas tanto al tipo de información que debe presentarse como al dominio de aplicación del sistema; por ejemplo, la iconografía utilizada para describir un elemento objetivo en el display es diferente en un sistema de radar de propósitos militares que en un radar de monitoreo de tráfico aéreo de uso civil.

3 ADQUISICIÓN DE SEÑALES

El componente de adquisición fue diseñado con el objetivo de proveer funciones de adquisición de señales, conversión de datos y sincronización de procesos de lectura de información.

El componente se compone de drivers de control de los dispositivos de hardware involucrados y software de lectura de datos. Esta aplicación de lectura interactúa con los distintos componentes obteniendo aquella información relevante para el sistema y lleva a cabo todas las transformaciones de datos requeridas por el mismo. Es decir: todas aquellas señales analógicas que deban ser procesadas por el sistema son digitalizadas en este componente, por medio del control de placas conversoras analógico-digitales. Además, se implementan aquí todas las funciones de conversión de datos necesarias para que el componente de procesamiento reciba como entrada un conjunto de información estandarizada, según un formato definido previamente.

4 PROCESAMIENTO DE SEÑALES

4.1 Descripción General

El componente de procesamiento fue diseñado de modo de poder analizar las diferentes señales de entrada en tiempo real, calculando los valores correspondientes a los «elementos» u «objetivos» que componen las diferentes presentaciones y enviando dicha información al componente de visualización.

Debe tenerse en cuenta que para calcular los datos correspondientes a algunos objetivos, es necesario analizar la información proveniente de varias señales. Además, la información proveniente de algunas señales debe ser utilizada en el cálculo de varios objetivos; razón por la

cual, cualquier cambio en las señales de entrada implica recalculer los datos correspondientes a todos los objetivos afectados por dicho cambio.

Los datos de entrada recibidos desde el componente de adquisición y los datos de salida enviados al componente de visualización son transmitidos por estructuras de comunicación denominadas «SignalPipe», implementadas por medio de sockets de protocolo TCP. Estas estructuras fueron implementadas con motivo de proveer una interface estándar para la comunicación entre los componentes del sistema.

4.2 Implementación del Componente

El componente de Procesamiento del sistema (Véase Figura 2) se halla compuesto de:

- Una estructura de almacenamiento de información referida al conjunto de señales de entrada especificado.
- Un conjunto de algoritmos de cálculo de todas las propiedades y objetivos que componen las presentaciones definidas.
- Una estructura de dependencias entre señales de entrada y objetivos de salida.
- Referencias a las estructuras de comunicación con el componente de adquisición y el componente de visualización.

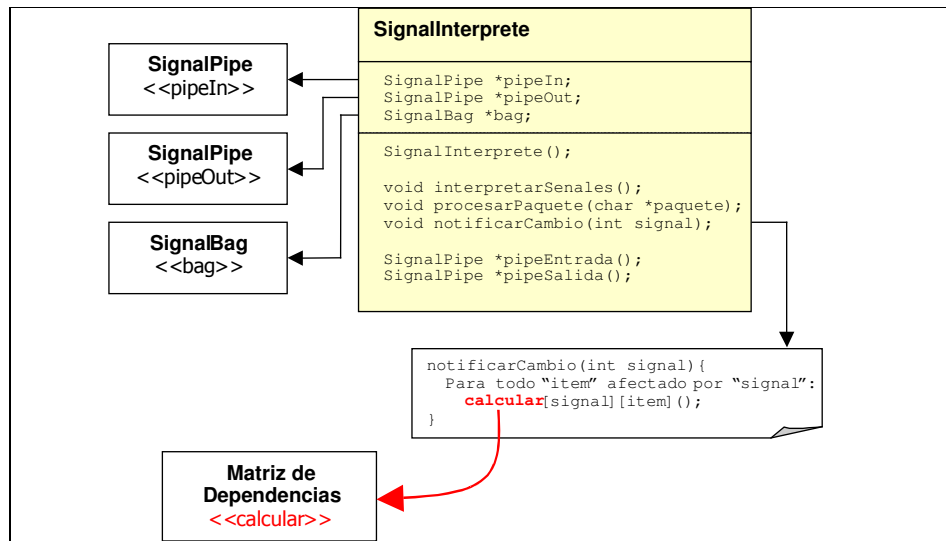


Figura 2: Diagrama de Implementación del Componente de Procesamiento

La estructura de almacenamiento de señales, implementada en la clase «SignalBag», mantiene en tiempo de ejecución toda la información necesaria de las distintas señales. Cada vez que se recibe un cambio correspondiente a una de sus señales, se actualiza la estructura y se notifica a los objetivos afectados. Internamente esta estructura posee un buffer de almacenamiento para cada señal. Estos buffers son de longitud variable –según las características propias de cada señal- y pueden ser actualizados y recorridos fácilmente, tanto a los efectos de incorporar los nuevos datos como de utilizar los datos ya almacenados para calcular la información de visualización.

El proceso de notificación consta de la invocación de las rutinas de cálculo de los objetivos afectados y el envío de los datos actualizados al componente de visualización. Para poder llevar a cabo esta tarea, el componente cuenta con una estructura donde se consignan qué objetivos dependen de qué señales. Esta estructura se denomina «Matriz de Dependencias» y contiene referencias a los algoritmos de procesamiento correspondientes a los distintos objetivos. Estos algoritmos son ejecutados por el método «calcular».

Las referencias a las estructuras de comunicación con los componentes de adquisición y visualización son denominadas «pipeIn» y «pipeOut», respectivamente.

La clase «SignalInterprete» es la clase principal del componente e implementa el mecanismo de procesamiento de las señales. El mismo es realizado por el método «interpretarSenales» (Véase Figura 3).

El método «interpretarSenales» implementa la función principal del componente; la cual recibe del pipe de entrada «pipeIn» los cambios en las señales. Para cada paquete recibido, se invoca al método «procesarPaquete», que actualiza los valores de las señales en la estructura «bag» e invoca las funciones de cálculo de los objetivos afectados por el cambio, las cuales notifican sus resultados al pipe de salida «pipeOut».

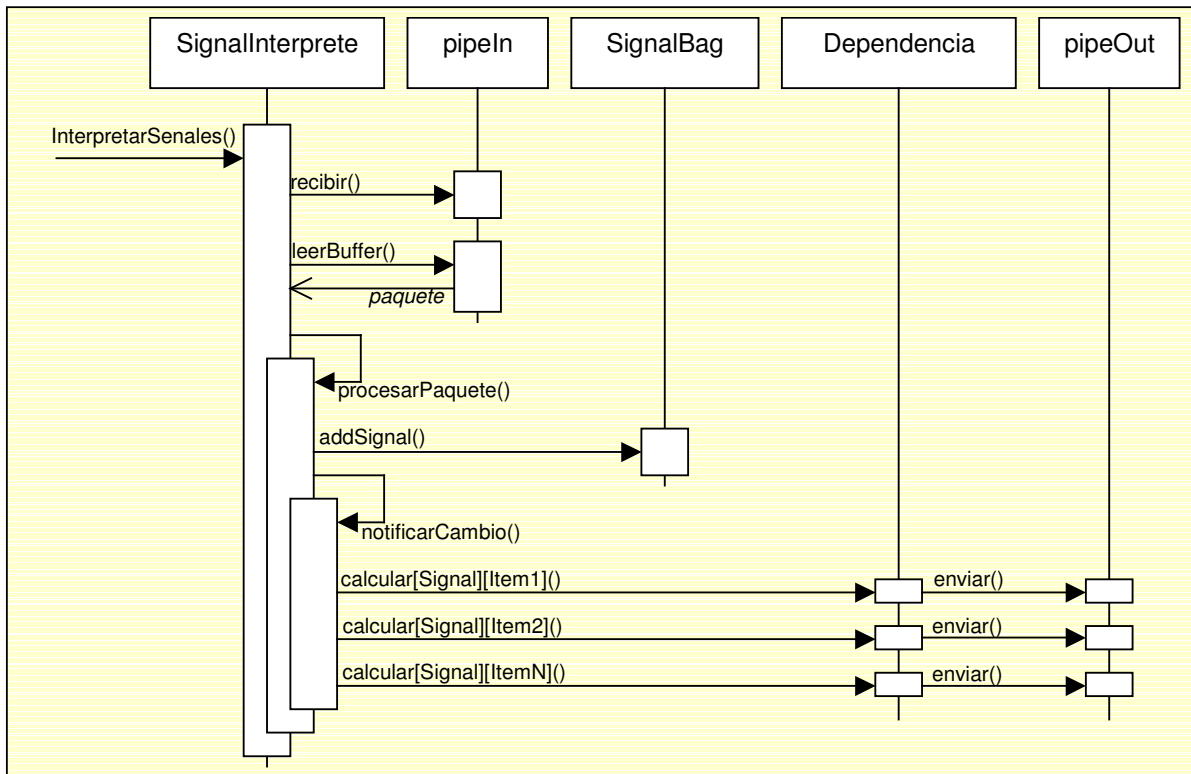


Figura 3: Diagrama de interacción del método «interpretarSenales»

4.3 Envío y Recepción de Datos

El envío y recepción de información con los componentes de adquisición y visualización se lleva a cabo respetando las convenciones definidas por las estructuras de comunicación de tipo «SignalPipe»:

- Los mensajes deben ser textuales y deben respetar un formato de ternas ELEMENTO-ATRIBUTO-VALOR.
- En algunos casos los mensajes que deben enviarse corresponden a duplas ELEMENTO-VALOR, para lo cual se adjuntará un valor cero (“0”) al tercer campo.

En el caso de los mensajes recibidos por el componente se utiliza el formato de duplas, donde el campo ELEMENTO se corresponde con un identificador de señal de entrada y el campo VALOR se corresponde, naturalmente, con el valor numérico de la misma.

En el caso de los mensajes enviados al componente de visualización, en cambio, se utiliza el formato de ternas, donde el campo ELEMENTO se corresponde con un identificador del elemento

objetivo en la presentación, el campo ATRIBUTO se corresponde con una propiedad del elemento objetivo y el campo VALOR con el valor numérico asociado.

5 VISUALIZACIÓN DE INFORMACIÓN

5.1 Descripción General

Resulta importante señalar que la unidad de visualización multifunción (MFD) debe componerse de, al menos, una “vista” o “presentación”. Entendiendo por vista un conjunto de información gráfica o textual que se presenta al usuario mediante un formato determinado oportunamente.

El componente de visualización fue diseñado de modo de poder soportar varias vistas en paralelo, con diferentes formatos y convenciones, pudiendo las mismas ser modificadas o reemplazadas de manera flexible. A tal efecto, se definieron dos elementos fundamentales: «Ítem graficable» y «Presentación».

Un Ítem graficable es cualquier elemento gráfico o información textual que deba ser presentado en el MFD; por ejemplo: un contacto, la regla de escala, la información de modo de barrido del radar, el fondo de la ventana donde se muestran los demás ítems, etc.

Una Presentación es básicamente un conjunto de ítems graficables. Una presentación puede ser una gráfica sinusoidal que describa los valores de una determinada señal del RADAR, un diagrama de la zona explorada por el radar o bien una lista de datos numéricos que informen al operador información asociada con un objetivo. Todos estos ejemplos pueden ser presentaciones diferentes de una misma situación. Cada tipo de presentación puede ser de mayor o menor utilidad según sea la necesidad. En todo caso, el presente diseño contempla la posibilidad de desarrollar cualquier tipo de presentación de manera fácil y eficiente.

5.2 Implementación del Componente

El componente de visualización fue desarrollado con el objetivo de poder manipular eficientemente información gráfica y textual (Ítems graficables) organizada en forma de presentaciones, las cuales deben ser implementadas en función de formatos de configuración definidos. Además, cabe aclarar que tanto los valores como las propiedades o parámetros visuales (color, tamaño, etc.) de los ítems graficables deben actualizarse en tiempo real.

Estas funcionalidades fueron implementadas de la siguiente manera:

- En primer término, se desarrolló la clase «VistaRadar» con el objeto de resolver todas las funciones propias de una presentación.
- Los ítems graficables fueron modelados por medio de la clase «ItemGrafico».
- Los formatos de las presentaciones son especificados a través de archivos de configuración, de forma tal de poder alterar el aspecto de la presentación sin necesidad de modificar o, incluso, recompilar el sistema.
- Finalmente, los cambios en la información presentada son recibidos a través de una estructura de comunicación con el componente de procesamiento del sistema (Véase sección 4.3).

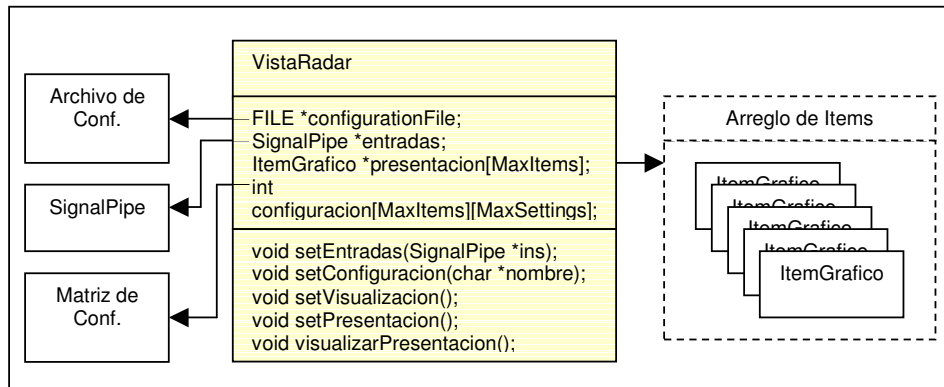


Figura 4: Diagrama de Implementación del Componente

La clase «VistaRadar» (Véase Figura 4) setea el formato de visualización de la vista descargando la información desde un archivo de configuración definido previamente. El objetivo del archivo de configuración es definir los formatos propios del sistema de representación y la iconografía a utilizar.

En tiempo de ejecución, la configuración de la presentación se almacena en una estructura interna denominada «Matriz de Configuración»; la cual almacena también información de los cambios que producen en el presentación en función de los distintos cambios en el estado del sistema (Por ejemplo: cambio en el modo de operación del radar, detección de un objetivo específico, etc.). Además, también es posible cambiar el formato de representación en tiempo de ejecución, para lo cual sólo debe actualizarse esta Matriz a partir de la información proveniente de otro archivo de configuración.

Los ítems graficables se encuentran en una estructura de tipo arreglo denominada «presentación» (Véase Figura 4). La misma es seteada a partir de la Matriz de Configuraciones y, conforme se obtienen los datos de las actualizaciones de los distintos ítems graficables (Por ejemplo: cambios en la localización de un contacto, cambio en el modo de operación del radar, etc.) se modifica, de modo tal de proveer una representación interna a la vista respecto de la situación a ser presentada en el MFD.

5.3 Implementación de Presentaciones

Una presentación se compone de un conjunto de elementos objetivos denominados “ítems graficables”. Cada ítem es representado en el sistema como una instancia de la clase abstracta «ItemGrafico».

La clase «ItemGrafico» define el comportamiento y las propiedades que caracterizan a un elemento de una presentación. Resulta importante señalar que para construir una presentación debe diseñarse una clase para cada tipo de ítem que la compondrá y que las mismas deben ser subclasses de «ItemGrafico». De esta forma, cada subclase implementa el comportamiento definido por la superclase y se obtiene un mecanismo unificado de tratamiento de los distintos elementos gráficos que conformarán las presentaciones del sistema.

Cada ítem graficable se define según un conjunto de propiedades comunes a todos (Por ejemplo: nombre, posición en el eje X, posición en el eje Y, color, etc.), cuyos valores se inicializan en función del formato de representación establecido en la Matriz de configuración (y ésta a su vez del contenido del archivo de configuración correspondiente) y se actualizan en tiempo real en función de la información de cambios provenientes del componente de interpretación.

Dado que muchos de los elementos a presentar en el MFD constituyen información textual, se definió también una clase abstracta que define el comportamiento específico de un ítem graficable textual. Esta clase se denomina «TextoG».

5.4 Configuración de Presentaciones

El objetivo del mecanismo de configuración a través de archivos es brindar flexibilidad a la manipulación de formatos de presentaciones. Este mecanismo permite que el sistema reciba toda la información necesaria para mostrar una presentación por pantalla desde un archivo independiente del sistema mismo (Véase Figura 5). De esta forma, si se respetan las convenciones adecuadas, es posible modificar archivos de configuración sin necesidad de recompilar el sistema.

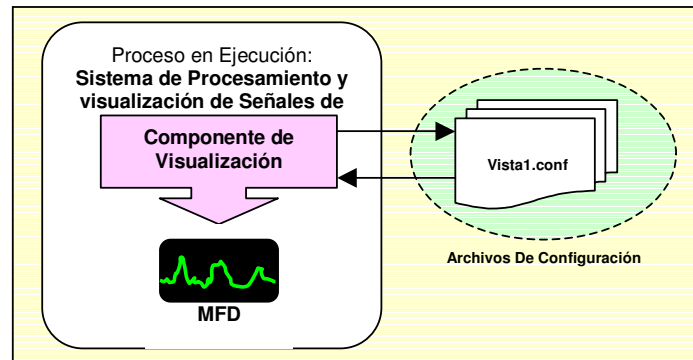


Figura 5: Interacción del Sistema con los Archivos de Configuración

Para cada representación que se desee proveer desde el sistema es necesario codificar un archivo de configuración asociado. Los archivos de configuración incluyen una definición clara de los distintos ítems graficables que incluye una determinada presentación, así como también los valores correspondientes a sus propiedades internas.

Estos ítems o elementos de las presentaciones permiten definir vistas que se adecuen a las necesidades específicas de cada dominio de aplicación. Por ejemplo, un ítem que debe definir cada presentación es el fondo de la misma, el cual puede ser tanto un color como un mapa o una fotografía satelital, lo que sirve tanto a los fines de un radar de tráfico aéreo como a radar meteorológico. Además, el sistema provee la capacidad de presentar en pantalla información textual, en diferentes tipografías y tamaños; gráficos elementales, como ser líneas o círculos; e imágenes en distintos formatos, de modo de poder identificar un cierto objetivo mediante una representación gráfica del mismo (una imagen de un barco, una nube, una bandera o cualquier otro símbolo distintivo).

6 IMPLEMENTACIÓN

A modo de caso de estudio, se implementó un sistema de software según las características descritas en las secciones anteriores.

El sistema ha sido programado íntegramente en lenguaje ANSI/C. en función de las características de administración de procesos y prestaciones para sistemas de tiempo real, ha sido implementado sobre un sistema operativo Linux Sarge.

6.1 Adquisición de Señales

En lo que respecta a la plataforma tecnológica utilizada, se implementaron controladores para dos dispositivos de adquisición de señales analógicas y digitales. La primera es una placa de tipo PCI provista por la empresa *Measurement Computing*, modelo PCI-DAS-4020/12 (Véase Figura 6-A). La segunda es una placa de tipo PCI provista por la empresa *Eagle Technology*, modelo PCI-703S (Véase Figura 6-B).

La placa PCI-DAS-4020/12 posee 4 canales de entradas analógicas, 2 canales de salidas analógicas y 24 canales de entrada/salida digitales. Esta placa opera con una resolución de 12 bits y permite muestrear datos a una frecuencia de hasta 20MHz.

La placa PCI-703S posee 8 canales de entradas analógicas, 2 canales de salidas analógicas y 8 canales de entrada/salida digitales. Esta placa opera con una resolución de 14 bits y permite muestrear datos a una frecuencia de 400KHz.

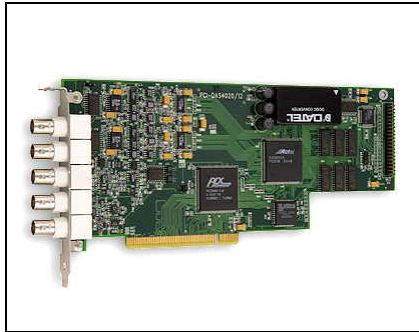


Figura 6-A: Placa PCI-DAS-4020/12

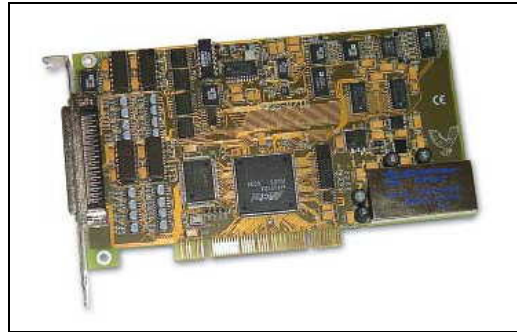


Figura 6-B: Placa PCI-703S

Con estas placas es posible obtener una amplia variedad de señales, de diferentes frecuencias y características, de modo de poder poner en práctica diferentes algoritmos de interpretación.

6.2 *Procesamiento de Señales*

Los algoritmos de interpretación de señales utilizados hasta el momento permiten efectuar operaciones de filtrado de ruido, identificación de posibles objetivos y cálculo de distancias de los objetivos identificados. Actualmente, se encuentran en etapa de estudio algoritmos de seguimiento de objetivos y reconocimiento automático.

6.3 *Visualización de Información*

Diferentes presentaciones han sido programadas y puestas en práctica. La mayoría de las mismas son variaciones de las presentaciones tipo B, las cuales ubican en el display los distintos objetivos identificados en una representación cartesiana según la información de distancia (Eje Y) y azimuth (Eje X).

Además, se implementaron presentaciones tipo A, las cuales muestran en el display una representación cartesiana según la información de amplitud de las señales recibidas por la antena (Eje Y) y distancia (Eje X) [1].

6.4 *Estado Actual*

El estado de avance actual del sistema permite adquirir, interpretar y visualizar información de un sistema de radar, basándose en señales producidas por un generador. De esta forma, a un nivel de simulación es posible estudiar diferentes algoritmos de procesamiento así como implementar diferentes mecanismos de presentación de información.

7 **CONCLUSIONES Y TRABAJO FUTURO**

Según las pruebas efectuadas hasta la fecha, es posible concluir que el sistema desarrollado puede ser aplicado con éxito en diferentes sistemas de radar existentes, con el objetivo de implementar cambios, actualizaciones o mejoras en los mismos.

La arquitectura de software propuesta resultó satisfactoria para garantizar el cumplimiento de los requisitos funcionales y no funcionales contemplados. De todos modos, resulta crítico aplicar este sistema en diferentes plataformas tecnológicas, de modo de poder probar las prestaciones funcionales del mismo en distintos dominios de aplicación.

A corto plazo, es de suma importancia implementar diferentes algoritmos de procesamiento de señales. En primer término, deben desarrollarse algoritmos de reconocimiento automático y de seguimiento de objetivos y/o trayectorias.

El desarrollo de nuevas presentaciones resulta también un aspecto de singular importancia, pues permite adaptar el sistema actual a nuevos y diferentes dominios de aplicación.

8 REFERENCIAS BIBLIOGRÁFICAS

- [1] Hovanessian, S. A. Radar Detection and Tracking Systems. Artech House, Inc. 1973.
- [2] Sha, L. *et al.* Evolving Dependable Real-Time Systems. Technical Report CMS/SEI95 -TR-005, CMU, Software Engineering Institute. 1995.
- [3] Perry, R. *et al.* Trellis Structure Approach to Multitarget Tracking. Adaptive Sensor Array Processing Workshop, Marzo de 1999. MIT Lincoln Laboratory.
- [4] Johnson, J. R. y Johnson, R. W. Challenges of Computing the Fast Fourier Transform. Optimized Portable Application Libraries (OPAL) Workshop, Kansas City. Junio de 1997.