

COMPARACIÓN DE CORBA Y X WEB EN EL DESARROLLO Y DESEMPEÑO DE UNA INTERFAZ GRÁFICA MULTIUSUARIO

Bencomo N.*, Correa E.*, Matteo A.[&]

{nbencomo, ecorrea}@blues.ciens.ucv.ve, amatteo@isys.ciens.ucv.ve

SimLab, Centro IOMMA*,

Laboratorio TOOLS, Centro ISYS[&],

Escuela de Computación, UCV,

Apartado Postal 47002, Caracas, Venezuela

Resumen

En este documento se presentan los resultados de una evaluación comparativa entre CORBA y X Window para la Web (X Web) en el desarrollo y desempeño de una Interfaz Gráfica Multiusuario para un Sistema en Tiempo Real. La evaluación se sitúa con respecto a las etapas de diseño, implementación y desempeño de una Interfaz Gráfica Multiusuario usando CORBA y X Web. Entre los aspectos considerados para la comparación de estas tecnologías están la complejidad en el desarrollo del software, aspectos económicos, extensión, mantenimiento y tiempos de respuesta de la Interfaz Gráfica. Los resultados obtenidos permiten entender que tanto CORBA como X Web presentan pros y contras dependiendo del escenario de operación en la que se desenvuelva una aplicación como la tratada.

Palabras Claves: CORBA, Despliegue Gráfico Distribuido, Interfaces Gráficas Multiusuario, Objetos Distribuidos, X Web, X Window.

INTRODUCCION

La interfaz gráfica utilizada en este estudio permite la interacción de diferentes usuarios con el Sistema de Modelos de Transporte de Líquidos por Tuberías SMTL, sistema que modela el transporte de crudos a través del oleoducto entre los estados Apure y Barinas en Venezuela. El desarrollo del SMTL comenzó al inicio de los 90, y sus requerimientos a nivel de su uso potencial estuvo limitado a la tecnología de la época. Con los nuevos avances tecnológicos, tanto en hardware como en software, se planteó una redimensión de la Interfaz Gráfica de Usuario IGU del SMTL de forma que el sistema pudiese ser consultado por diferentes usuarios, utilizando la tecnología Web. Al investigar acerca de las nuevas tecnologías que pudiesen servir como bases para el desarrollo de la nueva interfaz gráfica multiusuario, se encontró que CORBA (Common Object Request Broker Architecture) permite la construcción de interfaces gráficas multiusuario basadas en objetos distribuidos que pueden acceder a la misma información desde diferentes plataformas; y que por su parte, X Web permite también que diferentes interfaces gráficas sean desplegadas sobre plataformas heterogéneas en una forma distribuida. De esta forma, aún cuando estas tecnologías fueron concebidas para propósitos diferentes, CORBA y X Web pueden ser usadas para este mismo fin.

El estudio del comportamiento de cada una de estas tecnologías en la ejecución de una aplicación gráfica multiusuario de un sistema en tiempo real es el área de investigación que motiva este trabajo.

CORBA es una especificación estándar de una arquitectura de objetos distribuidos, respaldada por el consorcio OMG (Object Management Group), quien con mas de 700 miembros promete una larga y notable existencia para los años venideros. CORBA ofrece la posibilidad de construir aplicaciones basadas en objetos distribuidos, lo que permite la realización de los objetivos planteados: distintas interfaces gráficas que acceden a la misma información desde distintas plataformas. Por otro lado, X Window, tecnología con mucho más tiempo y con un gran respaldo, permite la posibilidad de desplegar distintas interfaces gráficas sobre diferentes plataformas en forma distribuida, con la facilidad del diseño e implementación de una aplicación que es ejecutada en forma local. En este caso el diseñador no tiene por qué preocuparse por la distribución, X se encarga de ello en tiempo de ejecución.

La aplicación desarrollada permite estudiar a CORBA y X Web como tecnologías utilizadas en el despliegue de las Interfaces Gráficas de Usuario y un sistema en tiempo real, en lo que se refiere a complejidad en el desarrollo del software, aspectos económicos, extensión, mantenimiento y tiempos de respuesta de la Interfaz Gráfica.

Este artículo consta, además de la Introducción y las Conclusiones, de tres secciones: la sección I presenta una muy breve descripción del sistema en tiempo real a modificar. Luego, en la Sección II se presenta un resumen del desarrollo de la interfaz gráfica multiusuario obtenida y finalmente, en la Sección III, se presentan los criterios y resultados de la comparación de CORBA (utilizando la implementación Orbix) y X Web como los responsables de la transferencia de información entre el sistema en tiempo real y el usuario.

I BREVE DESCRIPCIÓN DEL SMTL

La primera versión de SMTL fue desarrollada en UNIX, los modelos matemáticos del sistema fueron implementados en C y FORTRAN, y la Interfaz Gráfica de Usuario original usa X Window, Motif y C. La Figura 1 muestra la arquitectura original del SMTL y la interacción entre sus módulos.

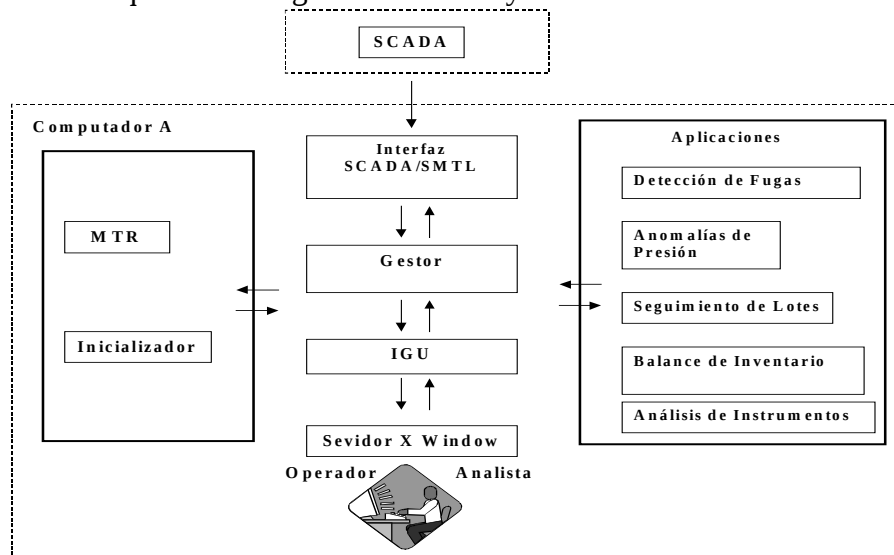


Figura 1: Arquitectura del SMTL con la IGU Monousuario

Se describe brevemente a continuación cada uno de los módulos del SMTL:

Módulo Inicializador: lleva a cabo la captura inicial de los datos para la inicialización del sistema.

Módulo Interfaz SMTL/SCADA: intermediario entre el SMTL y el SCADA (Supervisory Control and Data Acquisition).

Módulo de Tiempo Real (MTR): calcula el estado del flujo entre dos adquisiciones seriales de datos.

Módulo de Aplicaciones: grupo de subsistemas cuyo propósito es el estudio de toda la información obtenida al resolver los modelos matemáticos e inferir acerca de la existencia de posibles problemas.

Módulo de la Interfaz Gráfica de Usuario IGU: permite al usuario seguir las tendencias de las variables que describen el comportamiento del crudo en el oleoducto. También permite desplegar mensajes de alarma acerca de anomalías en el sistema

II DESARROLLO DE LA INTERFAZ GRÁFICA MULTIUSUARIO

Los requerimientos para la nueva versión de la Interfaz Gráfica del SMTL fueron los siguientes:

- Permitir que diferentes e independientes usuarios accedan los datos del sistema. Estas interfaces gráficas pueden ser desplegadas en forma remota.
- Ofrecer a los usuarios la posibilidad de tener diferentes componentes gráficos desplegados en forma simultanea.
- Incorporar la tecnología Web a la Interfaz Gráfica del sistema.

Al desarrollar la nueva versión de la interfaz gráfica del SMTL se dejó el módulo Gestor del SMTL casi intacto, para ello se utilizaron envoltorios (*wrappers*) que permitieron aislar y reutilizar al código como a una aplicación *legacy*.

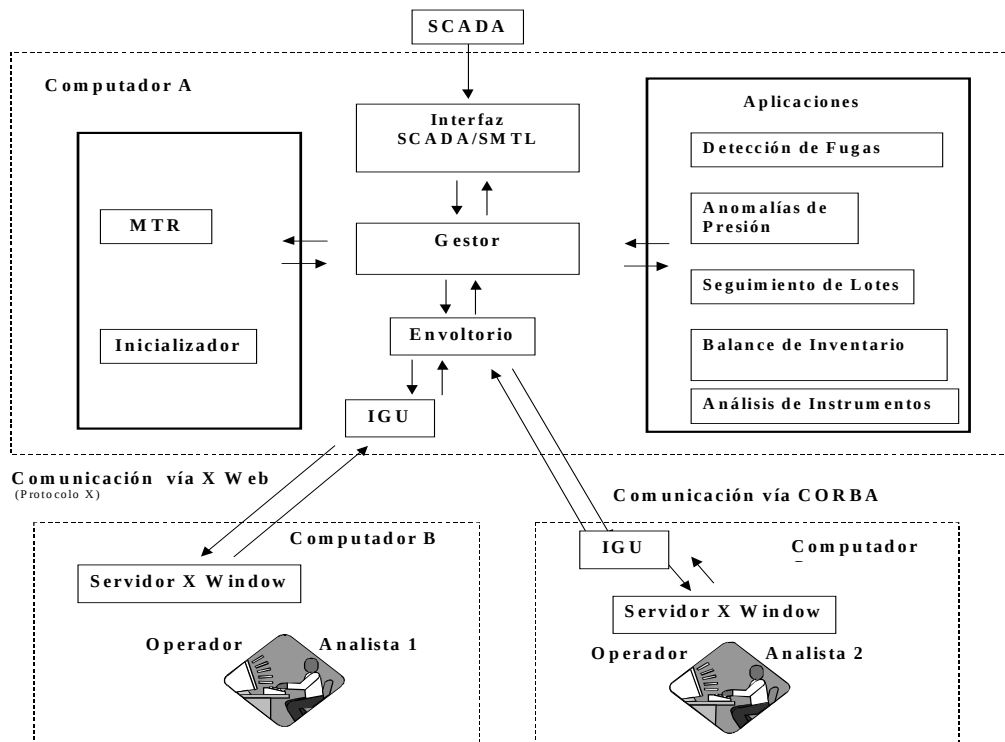


Figura 2: Arquitectura del SMTL con la IGU Multiusuario

La Figura 2 muestra la arquitectura del SMTL incorporando de la nueva versión de la IGU del SMTL. En el caso de la solución vía X Web, se tiene al usuario desplegando la aplicación en forma remota, sin embargo el proceso asociado a la IGU es ejecutado en forma local al SMTL. Aquí, la responsabilidad del despliegue remoto recae sobre el protocolo de Comunicación del Sistema X Window. En el caso de la solución vía CORBA, el proceso de la IGU es ejecutado en la máquina remota. Para este caso, todo lo relacionado a la comunicación entre las IGUs y los modelos matemáticos recae sobre los objetos distribuidos implantados con CORBA. Ambas soluciones pueden ser utilizadas, en forma individual o simultáneamente, en la versión actual del SMTL.

Basados en el método OOSE, se identificaron los casos de uso del SMTL a partir de los cuales se define la nueva interfaz gráfica multiusuario de este, estos casos de uso son los siguientes:

- **Conexión:** para realizar la conexión de usuario con el SMTL
- **Actualización:** para mantener actualizadas todas la IGUs conectadas al Sistema Matemático en Tiempo Real. Esta actualización se basa en un reloj de actualización, y se debe hacer cuando un tiempo predeterminado de actualización transcurre o se presenta un evento de alarma, todas las IGUs conectadas son actualizadas.
- **Visualización de Datos:** para satisfacer los requerimientos de visualización de datos..
- **Mensajes de alarma y/o advertencia:** para advertir a los usuarios acerca de situaciones de Alarma o Advertencia.
- **Modificación:** permite al administrador del SMTL la modificación del modelo. EL SMTL ejecuta dicha modificación y posteriormente el SMTL actualiza a todas las IGUS conectadas al SMTL.

Algunos de estos casos de uso representan formas de interacción genéricas de la IGU en las solicitudes de despliegues gráficos. Las solicitudes de despliegues gráficos pueden estar relacionadas a gráficas de perfiles de presión o temperatura, información de crudos en la tubería, etc. Estos casos de uso genéricos fueron definidos como casos de uso abstractos por lo cual debieron ser refinados en el proceso de desarrollo; en el contexto del objetivo de este artículo estos detalles no son relevantes.

En la etapa de análisis, se identificaron los objetos interfaz los cuales corresponden a la IGU, los objetos entidad como aquellos relacionados con el Gestor de los modelos matemáticos computacionales, y los objetos de control como los encargados de la comunicación entre los objetos interfaz y los objetos entidad, [Ben&al99], [Ben98].

En la etapa de diseño de la solución vía CORBA cada objeto de control de la etapa de análisis pasa a ser un bloque de control intermediario que permite la comunicación entre el sistema de modelos matemáticos y las IGUs. Cada bloque de control es dividido en dos objetos internos que solicitan y satisfacen los servicios entre sí. Estos bloques de control pueden convertirse en un bloque de control distribuido, cuyos objetos internos pueden estar en diferentes plataformas pero interactuando como si estuvieran en la misma máquina, cumpliendo así con el concepto de objetos distribuidos. Ver Figura 3.

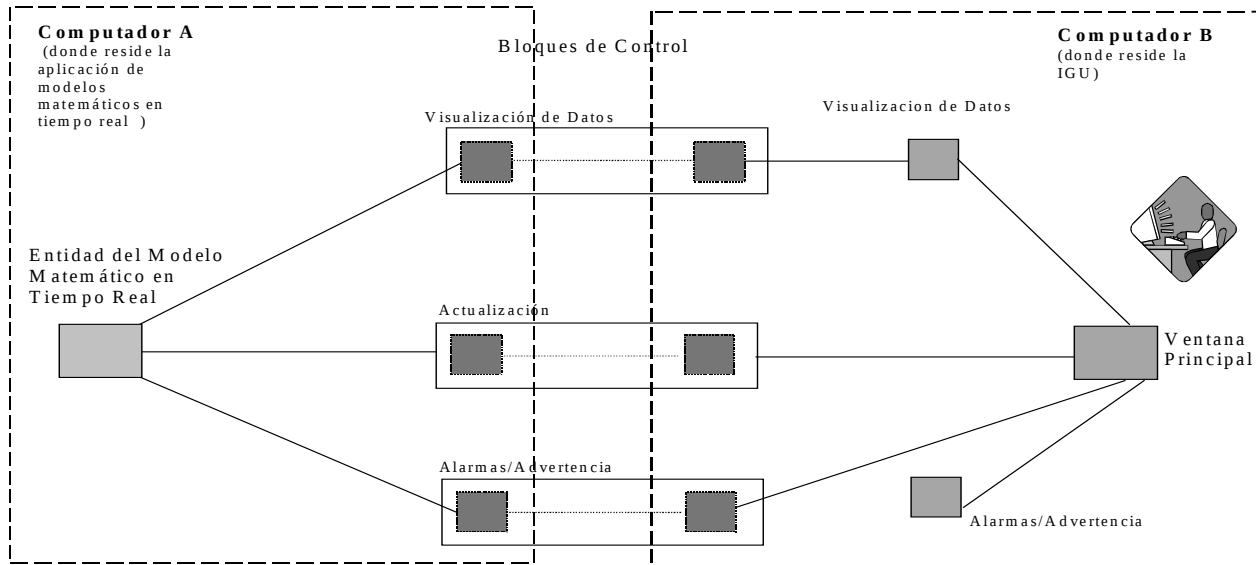


Figura 3: Solución CORBA

La utilización de X Web para el despliegue distribuido es presentado en la Figura 4. X Web no influyó en forma crucial en el proceso de desarrollo de la aplicación distribuida. Los procesos de las diferentes IGUs son ejecutados en la misma máquina donde reside el sistema en tiempo real. El Protocolo del Sistema X Window realiza la comunicación necesaria para efectuar el despliegue remoto.

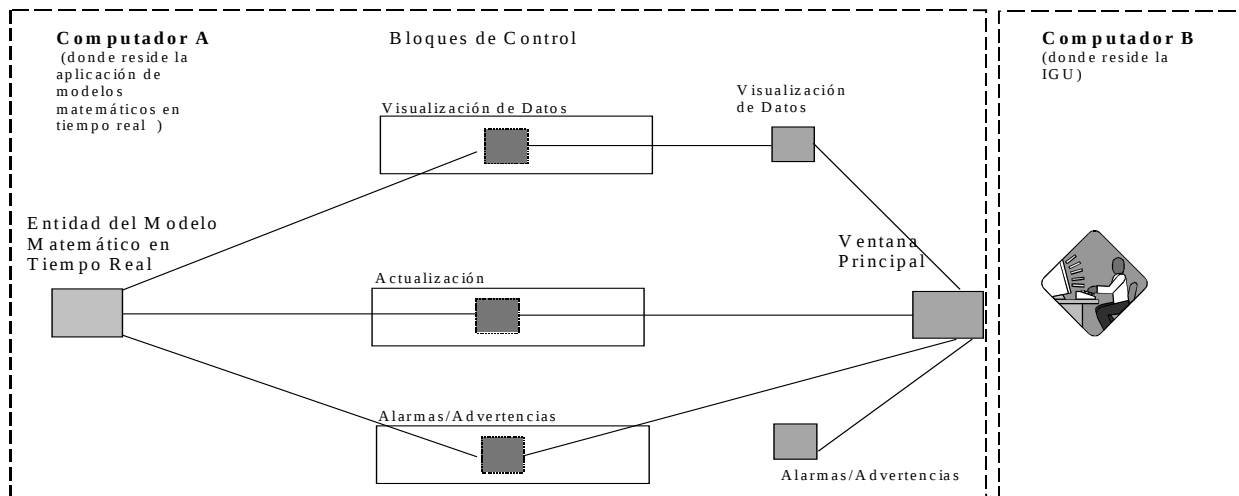


Figura 4: Solución X Web

Para la implementación de la IGU Multiusuario del SMTL, se utilizó la implementación comercial de CORBA IONA/Orbix 2.0 para C++, y como herramientas para los componentes gráficos X11R6.3 (para X Web) Motif 2.0, y XRT Widgets Vol. 11.

III EVALUACION

En este capítulo se presentan los criterios utilizados y los resultados obtenidos en el estudio comparativo entre CORBA (Orbix) y X Web como tecnologías utilizadas en el despliegue de las Interfaces Gráficas de Usuario para un Sistema de Modelos Matemáticos en tiempo real. El estudio está basado en cuatro criterios: complejidad en el desarrollo, costos, extensión y/o mantenimiento y rendimiento o desempeño de la interfaz gráfica remota.

III.1 CRITERIOS DE COMPARACIÓN

A continuación se describen los criterios de comparación utilizados:

- **Complejidad:** grado de dificultad asociado al desarrollo de la aplicación.
- **Costos:** costo del software obtenido en base a las horas/hombre y el costo de la tecnología utilizada.
- **Extensión y/o mantenimiento:** ventajas proporcionadas por cada uno de las tecnologías al realizar posibles extensiones y/o cambios del software a futuro.
- **Desempeño:** tiempo de respuesta cuando el usuario hace requerimientos de información gráfica a la IGU.

III.2 COMPARACIÓN

Las pruebas de evaluación fueron realizadas en las siguientes arquitecturas:

- SPARC Station 5 con Solaris 2.5, donde fue ejecutado el modelo, la IGU con despliegue remoto vía X Web.
- SPARC Station Ultra-4, con Solaris 2.6: donde fue ejecutada y desplegada una IGU remota al modelo, y fue desplegada la IGU ejecutada en forma local al modelo.
- Indy con IRIX 5.3: donde fue ejecutada la IGU con despliegue remoto vía X Web.

Las máquinas están conectadas a una red local con tráfico considerable a una velocidad de 10 Mbits/seg. La implementación de CORBA a utilizar es Orbix versión 2.2. La versión de X Web utilizada fue X11R6.3.

COMPLEJIDAD

Al utilizar el estándar CORBA por medio de una de sus implementaciones (Orbix), se experimentó un esfuerzo mucho mayor en el desarrollo del software, en relación al experimentado con X Web. Esto se debió fundamentalmente a la necesidad de especificar las interfaces IDL de los objetos distribuidos con CORBA, además de toda la programación asociada a la implementación de estos. Con X Web, el diseño e implementación de la aplicación puede ignorar aspectos asociados al despliegue distribuido en las máquinas donde se encuentran los usuarios remotos. De todo lo concerniente a estos aspectos de despliegue, son responsables los servidores X de las máquinas donde se encuentran los usuarios.

COSTOS

Las licencias por arquitectura de las implementaciones de CORBA presentan costos bastante elevados. Aunque existen muchos brokers de dominio público, generalmente estos no proveen los mecanismos necesarios para servicios específicos, como acceso a Sistemas de Bases de Datos y Sistemas de Seguridad. Estos servicios hacen que los fabricantes de brokers comerciales incrementen enormemente

sus precios. Por otro lado, el sistema X Window en cualquiera de sus versiones es de dominio público, aun cuando X es un sistema bastante robusto y continuamente actualizado por todas las compañías y empresas de prestigio que respaldan al Consorcio X.

Por otra parte, debido a la complejidad en el desarrollo de la aplicación, los costos en cuanto a la duración del desarrollo (horas/hombre), se pueden suponer más altos que los requeridos en un desarrollo con X Web.

Otra ventaja económica de X Web sobre CORBA, reside en el hecho de que una aplicación X Web, aun cuando ha sido desarrollada para plataformas UNIX, esta puede ser desplegada en otras plataformas, por ejemplo MS-Windows, sin necesidad de reprogramación. Las licencias de los servidores X para MS-Windows no son de dominio publico, pero en comparación a las versiones de CORBA para MS-Windows, son mucho más económicas.

EXTENSIÓN Y/O MANTENIMIENTO

A pesar de la complejidad asociada al desarrollo de una aplicación basada en CORBA, CORBA permite obtener un sistema con alto grado de modularidad y por consiguiente facilidad en posibles extensiones del software resultante. Para el caso de X Web, las facilidades de extensión y/o mantenimiento dependen de la modularidad con la cual el diseñador realice la aplicación.

DESEMPEÑO

Los tiempos de respuestas de la IGU al usuario utilizados para la evaluación del desempeño de la aplicación fueron los siguientes:

1. Tiempo del despliegue de una gráfica de perfil¹ (TR1): Intervalo de tiempo transcurrido desde el instante en que el usuario hace clic en una posición del oleoducto hasta que el despliegue del perfil asociado es realizado.
2. Tiempo ínter despliegue de una gráfica de perfil (TR2): Intervalo de tiempo transcurrido entre dos despliegues sucesivos de datos causados por el Sistema de Modelos para actualizar los datos mostrados al usuario. Estos despliegues se producen cada vez que el Scada proporciona nuevos datos -“Scan” -y el Sistema de Modelos produce nueva información para el(los) usuario(s) final(es).

A. TIEMPOS DE RESPUESTA TR1

Con la utilización de X Web se obtuvieron tiempos de respuesta TR1 bastante satisfactorios (ver Tabla 1). Aún cuando la media es de aproximadamente 15,7 segundos y puede parecer muy grande, es necesario considerar que los intervalos de tiempo “Interscan” de actualización de la información en el SMTL, son en promedio del orden de 28.37 segundos (ver Tabla 2). Al momento en que la IGU requiere información al Gestor de los Modelos, este esperará hasta tener los cálculos para el próximo Scan, antes de entregar la información solicitada. Si suponemos uniformidad sobre esta espera del

¹ Una gráfica de perfil consiste en una gráfica de una variable (por ejemplo presión, temperatura, densidad) en función del la progresiva o kilometro en el Oleoducto.

Gestor de los modelos, se tendría una media de 14.18 segundos, lo cual se acerca bastante al valor la media de TR1 (15,7), con lo cual el proceso realizado por X Web sería en promedio de 1 segundo.

<i>X Web</i>	
Media	15.713636
Mediana	15.45
Moda	14.9
Desviación estándar	1.9067038
Mínimo	12
Máximo	19.9
Nivel de confianza(95.0%)	0.8453852

Tabla 1: Estimados estadísticos básicos de TR1 para Web

<i>Inter-Scada</i>	
Media	28.373494
Mediana	28
Moda	28
Desviación estándar	0.5567991
Máximo	31
Nivel de confianza(95.0%)	0.12157962

Tabla 2: Estimados estadísticos básicos para los tiempos Interscan

La **Figura 5** muestra el histograma de frecuencias del tiempo de respuesta TR1 cuando se utiliza X Web, como se puede observar los tiempos de respuestas más comunes se encuentran alrededor de 15.5, 16.5 y 18.5.

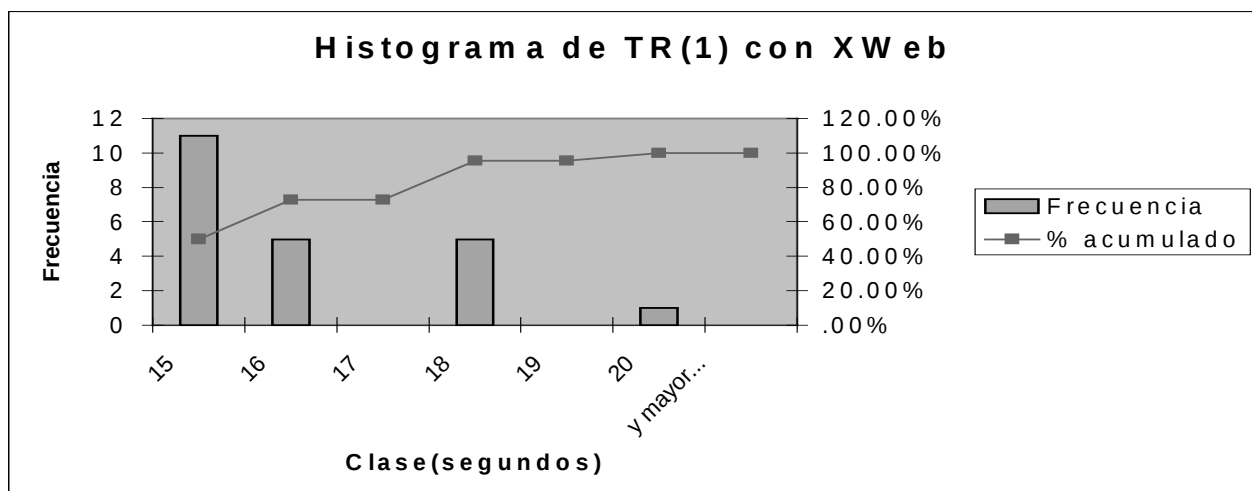


Figura 5: Histograma de frecuencias de TR1 con X Web

En cuanto a la solución vía CORBA y en base a nuestra experiencia obtenida con usuarios finales de Interfaces Gráficas remotas de un sistema en tiempo real, se constató la conveniencia de enviar información actualizada de los modelos al comienzo de la conexión de la IGU, cada cierto periodo de tiempo predefinido (por ejemplo cada 15 min.) y cada vez que se produjera un cambio significativo en el Oleoducto (por ejemplo un cambio de operación en el oleoducto). De esta manera se puede mostrar al usuario información de forma rápida, en el momento que este la solicitara, consultando los datos en forma local, es decir los últimos datos enviados por el Gestor.

Al incorporar esta estrategia, en la aplicación con CORBA, los TR1 resultaron bastante satisfactorios, ya que los tiempos transcurridos para el despliegues nunca fueron mayores de 1 segundo, obviamente, debido a que el acceso a los datos fue local.

B. TIEMPOS DE RESPUESTA TR2

En relación a los intervalos de tiempo entre actualizaciones de la información gráfica, producto de nuevos cálculos con información de nuevos Scans (TR2), su valor promedio fue de 29.35 segundos con un intervalo de confianza del 95% del orden de 0.72 segundos alrededor de esa media.

<i>Corba</i>	
Media (seg)	29.357142
Mediana	30
Moda	31
Desviación estándar	3.3207738
Máximo	37
Nivel de confianza(95.0%)	0.7206522

Tabla 3: Estadísticas básicas de TR2 utilizando CORBA

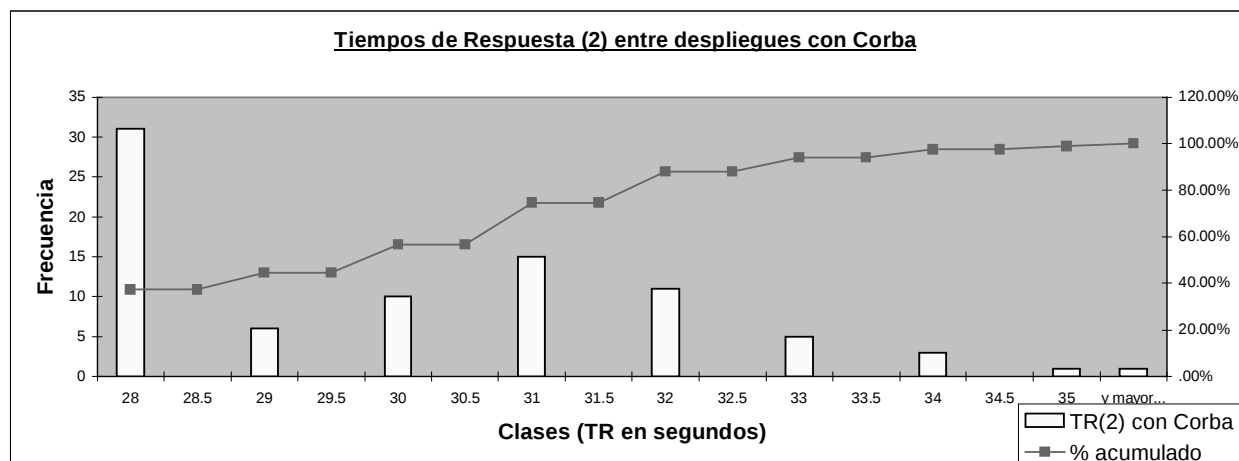


Figura 6: Histograma de frecuencias de TR2 con CORBA

Estos tiempos TR2 con CORBA son bastante aceptables, tomando en cuenta que los tiempos Interscans tienen una media de 28.37 segundos, con lo cual el tiempo medio empleado en transmisión y despliegue estaría por debajo de un (1) segundo.

En comparación con X Web, CORBA mostró una mejora poco significativa en lo que respecta a un usuario de una IGU remota de esta aplicación. (ver Figura 7).

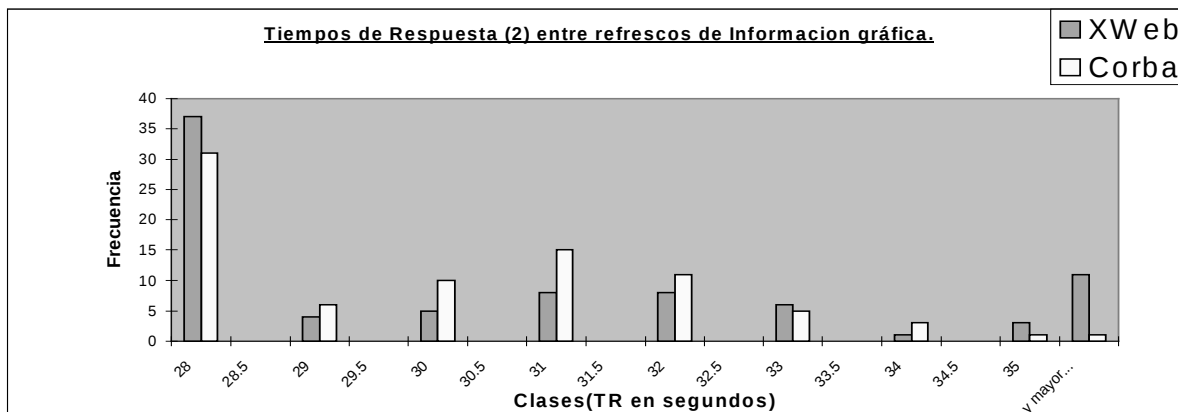


Figura 7: Tiempos de respuesta entre despliegues utilizando CORBA y X Web

Esta leve mejora de CORBA sobre X Web se debe a que el protocolo X transmite un volumen grande de información. CORBA por su parte envía solo la información asociada a los valores de los ejes x, y los datos del título y unidades, ya que el despliegue de la gráfica es local.

En estos despliegues sucesivos se observó que el protocolo X utiliza mecanismos de optimización para el despliegue de gráficas, se pudo ver que cuando una gráfica era desplegada y posteriormente actualizada, solo se refrescaban las áreas de las gráficas que habían cambiado. Lo anterior implica que el servidor X es una aplicación inteligente que optimiza el envío de información necesaria para los nuevos despliegues.

Comentarios generales:

Debido a que en los tiempos TR1 se tiene un comportamiento muy satisfactorio en base a la estrategia de actualización utilizada con Orbix (CORBA), y que no existe una diferencia notable entre los tiempos de respuestas entre X Web y Orbix en los tiempos TR2, se puede pensar que una estrategia combinada, Orbix para mejorar el TR1 y X Web para los inter despliegues. En el primer caso, la estrategia de actualización con Orbix reduce el tiempo de espera para el primer despliegue gráfico y X Web reduciría el esfuerzo de programación para los siguientes despliegues.

En relación a las IGUs que usan X Web, se puede decir que si se tienen varias IGUs locales al modelo, puede presentarse una dañina competencia entre los procesos de las IGUs y el proceso de los modelos matemático, ejecutándose simultáneamente en la misma máquina. Esto resultaría inconveniente en

cuanto a rendimiento de todo el sistema ya que los cálculos de los modelos deben ser en tiempo real. En este caso los tiempos de respuesta se podrían desmejorar significativamente dependiendo de la capacidad de procesamiento que se use para ese fin.

CONCLUSIONES

Junto al desarrollo de una Interfaz Gráfica Multiusuario para un Sistema de Modelos Matemáticos en Tiempo Real se realizó un estudio comparativo entre las tecnologías CORBA y X Web obteniéndose las siguientes conclusiones relevantes.

Respecto a la solución proporcionada por X Web:

La solución vía X Web es recomendable cuando la aplicación es ejecutada sobre una estación de trabajo de alto rendimiento con el fin de hospedar el Sistema en Tiempo Real y las diferentes IGUs que se estén desplegando remotamente. Si además esta máquina estuviese conectada a una red de alta velocidad, el desempeño del protocolo X es ideal.

X Web ofrece una solución menos compleja en diseño e implementación, debido a que el esfuerzo de la distribución de los despliegues gráficos recae sobre el Servidor y Protocolo X. Lo anterior y el hecho de que X Window es de dominio público para Unix y de muy bajo precio para MS-Windows, hace que X Web ofrezca una solución más económica.

Respecto a la solución proporcionada por CORBA:

Con CORBA, la IGU que es ejecutada en forma remota, puede ser actualizada en base a periodos de tiempo predefinidos o por cambios significativos en el oleoducto, permitiendo que las solicitudes de despliegues gráficos sean satisfechas en forma inmediata. En el caso en que esta rapidez de despliegue sea crucial, como puede ser el caso en otras aplicaciones de tiempo real, la solución ofrecida por CORBA es muy recomendable.

En una aplicación como la implantada, el esfuerzo de diseño necesario para la obtención de la solución que incluye CORBA, es mucho mayor que el realizado cuando sólo se utiliza X Web. Sin embargo, para la ejecución del software en redes WAN, donde la velocidad de transmisión es más lenta y por tanto puede resultar inconveniente, la solución basada en CORBA ofrece mayores ventajas.

Otras de las ventajas de diseño que permitió la implementación de CORBA, es el hecho de que el producto resultante es obligatoriamente modular, facilitando modificaciones y extensiones a futuro.

Una combinación interesante de estrategias es utilizar una implementación de CORBA para la realización del primer despliegue gráfico asociado a un requerimiento, y posteriormente utilizar X Web para los sucesivos actualizaciones de la gráfica

Finalmente, es importante resaltar que los resultados obtenidos con este trabajo permitieron aclarar que tanto CORBA como X Web presentan pros y contras dependiendo del escenario de operación en la que se desenvuelva una aplicación como la tratada.

BIBLIOGRAFÍA

- [Bar&al92] BARRIOS E., CORREA E., CREMA A., MARCANO P., MICARELLI M., MIRANDA G., PASTORIZA R., SEGOVIA H., SEPULVEDA G., VILLAMIZAR V., "SMTL: Computational System for Fluid Pipeline Transport Models", Proceedings II Congreso Internacional de Energía, Ambiente e Innovación Tecnológica, Roma, 1992.
- [Ben98] BENCOMO N., "Interfaz Gráfica Multiusuario Basada En Objetos Distribuidos Para Un Sistema En Tiempo Real" Tesis de Maestría, Postgrado en Ciencias de la Computacion, Facultad de Ciencias, UCV. Caracas, Venezuela, Noviembre, 1998.
- [Ben&al99] BENCOMO N, BLANCO A., CORREA E, MATTEO A.
"An Experience using OOSE for a Graphical Multi-User Interface based on Distributed Objects", Proceedings Volume 2 5th International Conference on Information Systems Analysis and Synthesis ISAS'99, Orlando, FL USA, julio, 1999.
- [Ben&al94] BENCOMO N, CORREA E,
"Design and Implementation of an Interactive Graphical Environment for Generic Mathematical Systems for Fluid Transportation through Pipelines", CLEI- PANEL 94, México, Abril, 1994.
- [Bro96] "X Web Information and Resource Center",
<http://www.broadway.com>, 1996.
- [IonP97] IONA Technologies,
"Programming Guide, Orbix 2", IONA Technologies PLC, 1997.
- [Jac&al93] JACOBSON, Ivar; CHRISTERSON, Magnus; JONSSON, Patrik; OVERGAARD, Gunnar "Object-Oriented Software Engineering: A Use case driven Approach", Addison Wesley, 1993.
- [Los&Matt97] LOSAVIO F., MATTEO A.,
"Use Case and multiagent models for object-oriented design of user interfaces", . Journal of Object-Oriented Programming, Mayo 1997.
- [Med97] MEHDI, Jazayeri
"Advances in Object-Oriented Software Engineering", Prentice Hall Object Oriented Series, Chapter 6, Distributed Software Design Techniques, 1997.
- [Orf&al94] ORFALI R., HARKEY D., EDWARDS J.,
"Essential Client/Server Survival Guide", Van Nostrand Reinhold, 1994
- [Ott&al96] OTTE R., PATRICK P., ROY M.,
"Understanding CORBA", Prentice Hall, 1996.
- [Rum&al91] RUMBAUGH, James; BLAHA, Michael; PREMERLANI, William; EDDY Frederick; LORENSEN, William
"Object-Oriented Modeling and Design", Digital Press, 1991
- [Sha93] SHATZ S.,
"Developing of Distributed Software", Macmillan Publishing Company, 1993.