

Ingeniería de Software de Sistemas distribuidos de Tiempo Real. Modelización y Evaluación de las Restricciones de Tiempo.

Ing. A.De Giusti¹, Lic. Hugo Ramón², A.C. Pamela Fernandez³, A.C. Andrea Artime³

*Laboratorio de Investigación y Desarrollo en Informática⁴
Departamento de Informática - Facultad de Ciencias Exactas
Universidad Nacional de La Plata*

Resumen

Se presenta un análisis de las extensiones en la metodología clásica de análisis y diseño de software, orientada a sistemas distribuidos de tiempo real (SDTR).

En particular se discute el modo de modelizar y verificar las restricciones de tiempo, utilizando Redes de Petri extendidas.

A modo de ejemplo concreto de aplicación se analiza el desarrollo de un sistema de software que detecta en tiempo real el corte de cables telefónicos troncales y utiliza información de GIS para alertar a los responsables de seguridad de la zona del corte. El área donde se experimenta el sistema es la Capital Federal.

Palabras clave: Tiempo Real. Ingeniería de Software. Redes de Petri.

¹ Inv. Principal CONICET. Profesor Tit. Ded. Excl., Dpto. de Informática, Facultad de Cs. Exactas, UNLP.
E-mail degiusti@lidi.info.unlp.edu.ar

²JTP Ded. Excl. LIDI. Dpto. de Informática, Facultad de Ciencias Exactas, UNLP.
E-mail hramon@lidi.info.unlp.edu.ar

³ Analista de Computación. Alumna avanzada de la Licenciatura en Informática. Dpto de Informática, Fac. de Ciencias Exactas, UNLP.

⁴ Calle 50 y 115 Primer Piso, (1900) La Plata, Argentina, Teléfono 54-21-227707
WEB: lidi.info.unlp.edu.ar

Introducción

El tratamiento de datos físicamente distribuidos en tiempo real es uno de los ejes del desarrollo de la tecnología de hardware y software en los últimos años. [Lapl98], [Levi90], [Sand89]. Los desafíos para la Ingeniería de Software y para los lenguajes de programación que impone la distribución física de equipos, datos y procesos constituye un área central de investigación en la Informática actual [Ward85], [Niel90].

La evolución tecnológica en el tratamiento de señales (locales o remotas) y en los sistemas de comunicaciones, así como los esfuerzos en la integración de recursos de hardware y software ha impulsado enormemente la especificación y el desarrollo de software para Sistemas Distribuidos de Tiempo Real [Burn96], [Josep96], [Härb94].

La Ingeniería de Software orientada a procesos en tiempo real ha requerido un importante esfuerzo de estudio teórico, sistematización metodológica y desarrollo de herramientas específicas [Geha86], [Davi93]. En particular el tema de la especificación y verificación de las restricciones de tiempo constituyen uno de los ejes principales de investigación, para el cual hay numerosos enfoques [Pete81], [Roze88].

En este caso se analiza la utilización de Redes de Petri extendidas en la especificación y verificación de un sistema distribuido de toma de decisiones en tiempo real, que integra comunicaciones automáticas, consulta a una Base de Datos central que contiene información de GIS e interfaz directa con hardware distribuido. Asimismo se discuten las herramientas de Ingeniería de Software utilizadas en el análisis y diseño del prototipo que ha sido básicamente el Trabajo de Grado de dos alumnas de la Licenciatura en Informática de la UNLP, ante un requerimiento de Telefónica de Argentina [Fern98].

Análisis del problema

Se ha estudiado y desarrollado un prototipo de sistema para la detección en tiempo real del corte de cables telefónicos y el comando de las acciones preventivas-correctivas correspondientes. En particular se analizan los aspectos de software, en particular el diseño y verificación de las restricciones de tiempo.

La descripción abstracta del problema es la siguiente:

- Existen un conjunto de centrales telefónicas en la Capital Federal y el Gran Buenos Aires, de las cuales salen conjunto de cables troncales con alrededor de 1000 pares telefónicos cada uno.
- Estos cables troncales se dividen sucesivamente en su recorrido, hasta llegar al usuario.
- Existe un nodo central de procesamiento que se puede comunicar en tiempo real (en principio por línea telefónica real o dedicada) con cada una de las centrales.
- El problema fundamental reside en que al producirse un corte y robo en un cable troncal o en uno de sus derivados (supongamos una rama de 250 pares telefónicos, por ejemplo), el costo de reparación/reposición es muy alto.
- Interesa obtener en tiempo real una detección del corte del cable, una estimación de la distancia y recorrido desde la central. Esta información se obtiene en cada central, a partir

de la medición del equipo de hardware dedicado que trasmite una señal y mide el retardo en recibir su eco a través del cable telefónico. Este tiempo es aplicado en la resolución de la ecuación de onda para determinar el recorrido de la señal y verificar si no hay un acortamiento en el mismo.

- Inmediatamente de producido el evento de corte de un cable, se interrumpe al procesador principal, donde reside un Sistema de Información Geográfica que dispone de los planos de la zona y tiene una Base de Datos con el recorrido del cableado.
- El procesamiento en el nodo principal permite disparar una alarma en tiempo real a la dependencia de seguridad de la zona y prevenir-reprimir el robo. *El tiempo de respuesta global no debe exceder los tres minutos.*

El prototipo desarrollado modeliza el problema y representa el comportamiento temporal del sistema mediante una Red de Petri extendida, permitiendo estudiar mediante trazas adecuadas, el comportamiento estadístico del sistema. En el desarrollo se utilizó el software de GIS Agemap 3.5 sobre el cual se armó el mapa de cableado telefónico de la Capital Federal.

Extensiones en el proceso de Análisis y Diseño

En el análisis y diseño del sistema se siguió la metodología estructurada extendida de Keller y Shumate [Shum92]. El lector interesado puede encontrar todos los detalles en el Trabajo de Tesis que se indica en [Fern98].

A los fines de visualización global del diseño se muestran los diagramas de Contexto de la Arquitectura, Diagrama de Flujo de la Arquitectura, Diagrama de flujo flat (FFD), Grafo de comunicaciones de Tareas (TCG) y el Diagrama de la Aquitectura Preliminar (SAD).

Diagrama de Contexto de la Arquitectura (ACD)

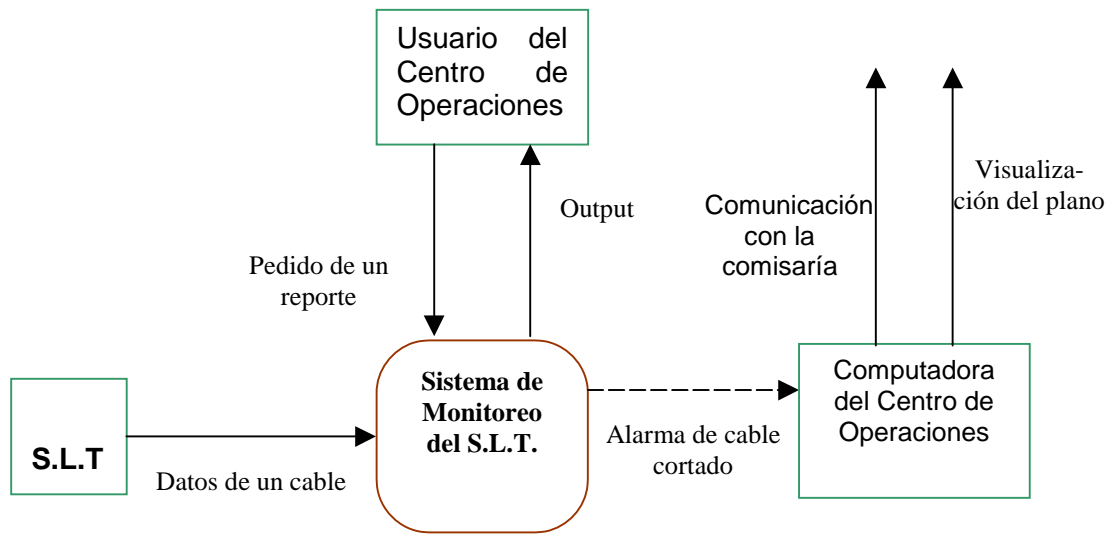


Diagrama de Flujo de la Arquitectura (AFD0)

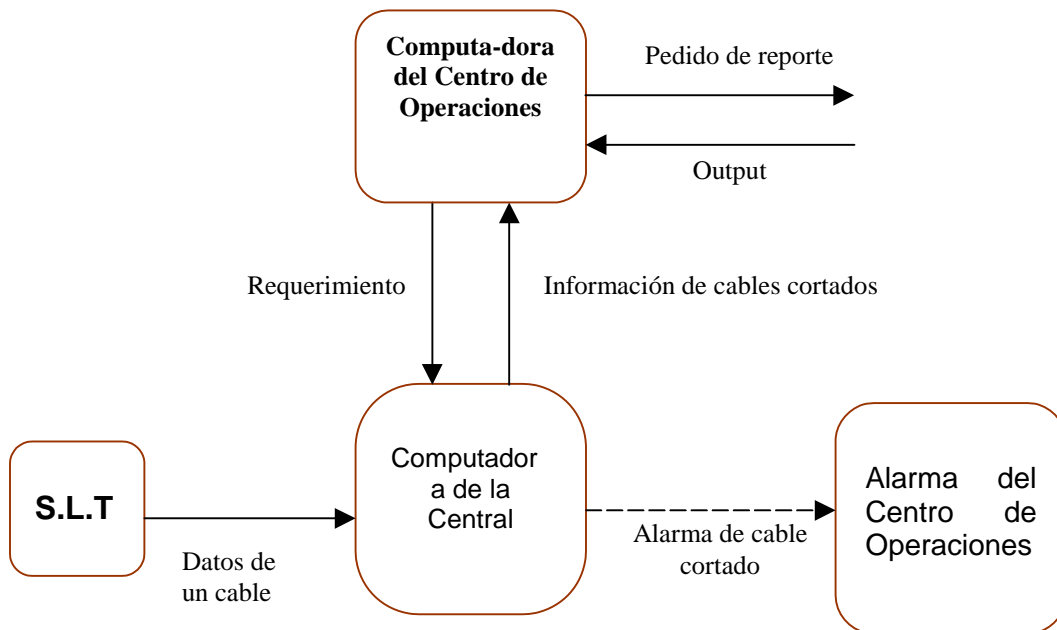


Diagrama de Flujo "flat" (F.F.D.)

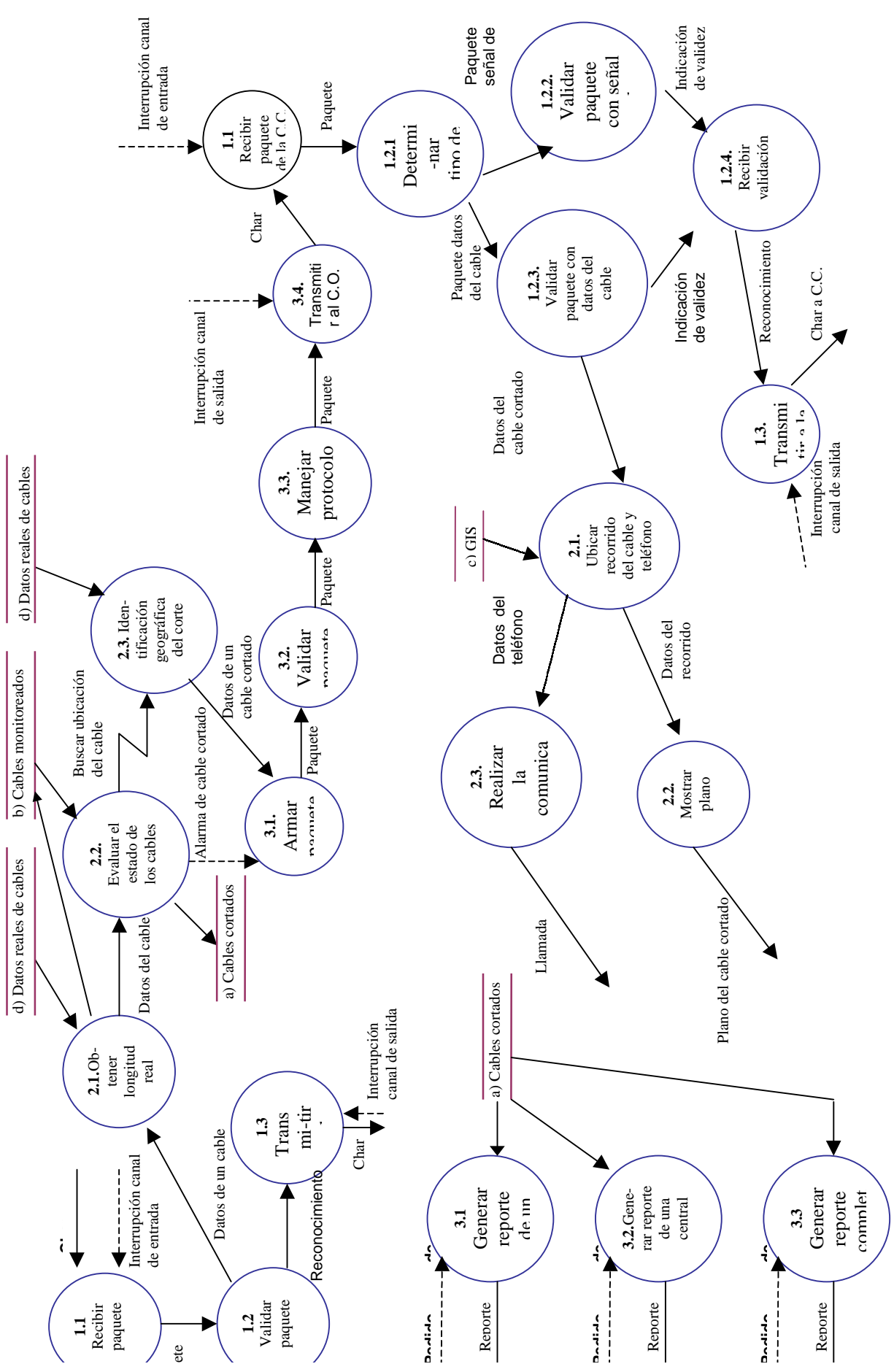
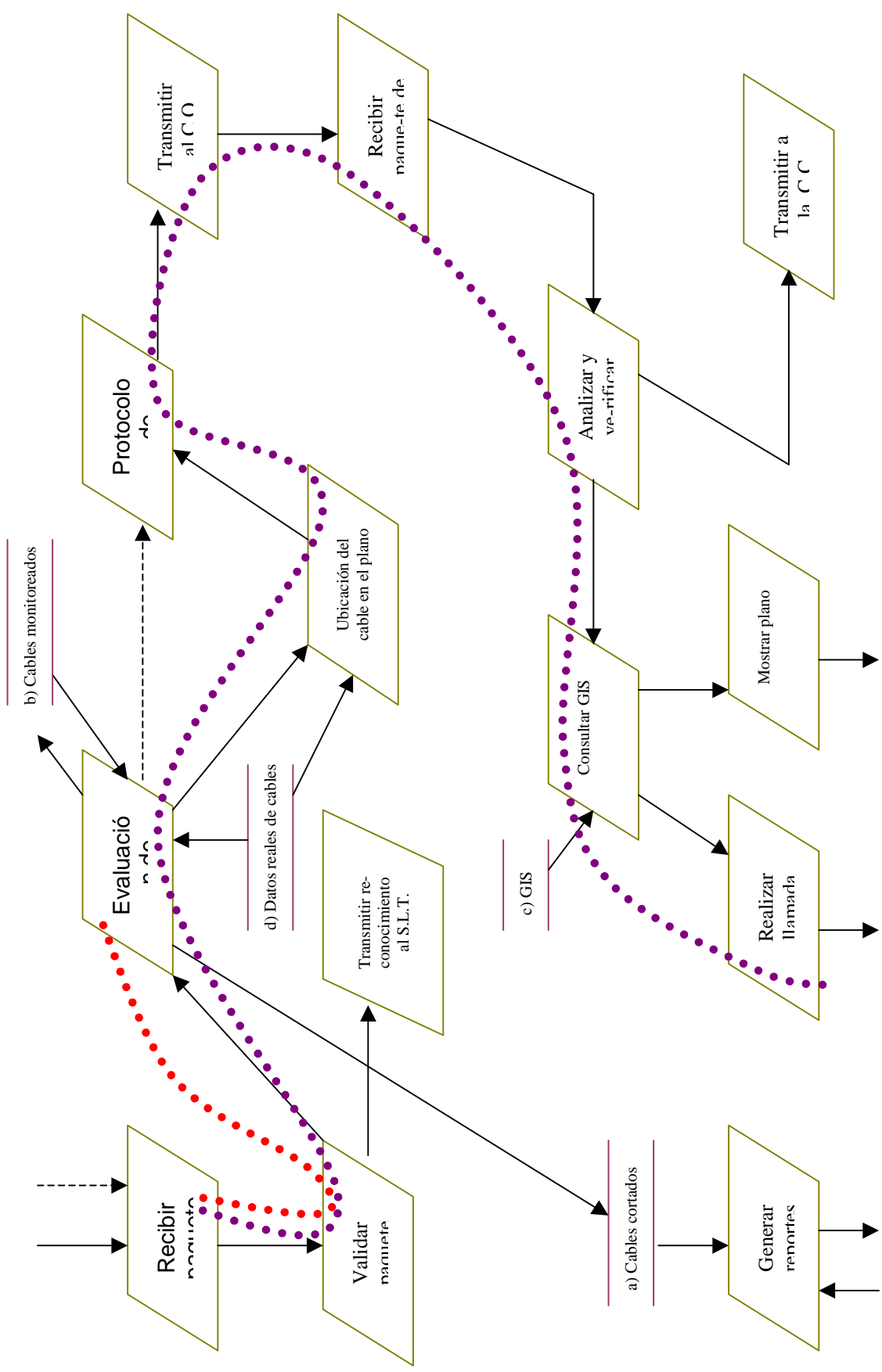


Gráfico de Comunicaciones de Tareas (TGC con Tiempo Crítico)



Modelización del tiempo. Redes de Petri extendidas.

Los procesos que involucran tiempos que deben chequearse en el prototipo se pueden resumir en:

- **Recibir la información del hardware de detección de cortes (SLT).**
El monitoreo de los cables se realiza sincrónicamente cada 1 segundo. El análisis en el sistema de software comienza al tener la lectura de la longitud del cable monitoreado.
- **Determinar si está cortado el cable leído.**
Una vez leída la longitud del cable se accede a un archivo local para obtener la longitud real del cable. Luego se compara y se determina si está o no cortado el cable.
Este módulo se realiza entre 11 y 14 mseg. Este dato fue calculado tomando una muestra de 50 cables cortados y sobre una microcomputadora standard tal como la que se podía utilizar en cada central (PC compatible tipo Pentium 166 con disco rígido IDE de 2.5 Gb).
- **Establecer la comunicación para transmitir los datos.**
Tomando como base un módem de 33600 bps, un par telefónico como medio de comunicación y una comunicación máquina-máquina, el tiempo para establecer la comunicación está determinado por el número de rings que se ejecutan hasta obtener respuesta (1 seg. por ring. aproximadamente), y el tiempo de negociación entre los dos módems, dependiendo de la velocidad de ambos módems. Al ser una comunicación entre dos máquinas, podemos considerar que la cantidad de rings estipulado es de uno. De acuerdo a las pruebas realizadas, se puede concluir que el tiempo medio para establecer la comunicación es 10 segundos. Este tiempo se incrementará en el caso de que la línea no esté disponible y haya que reintentar la comunicación. El incremento del tiempo será de 1 segundo aproximadamente, por cada reintento.
- **Transmisión de los datos.**
De acuerdo a la velocidad del módem mencionado anteriormente, y al tamaño de la información a ser transferida, el tiempo aproximado para la transmisión de los datos es menor a 1 segundo.
- **Buscar datos referentes a la comisaría y teléfono, y mostrar el plano.**
Este módulo comienza cuando se recibe la información de un cable cortado, el cual debe ubicarse geográficamente para identificar la comisaría correspondiente a la zona del corte. El tiempo va a depender del tamaño del mapa y la cantidad de recorridos trazados. Con el modelo de Agemap 3.5 de la Capital Federal, cargado con el recorrido de los cables troncales de interés trabajamos con una media de 10 seg y un tiempo máximo de 14 segundos.
- **Establecer la comunicación con la comisaría.**
Tomando como base un módem de 33600 bps, un par telefónico como medio de comunicación, los tipos de comunicación que se pueden establecer son: máquina-máquina, máquina-fax, máquina-hombre. En el primer caso el tiempo estimado es igual al del punto 3, teniendo en cuenta la mismas consideraciones y los mismos retrasos. En el segundo caso la cantidad de rings puede incrementarse lo que determina un aumento en el tiempo (siempre 1 segundo por ring con una media de 5 llamados antes de atender y distribución normal). En el tercer caso, al ser atendido el teléfono por una persona aumenta considerablemente el tiempo, ya que la cantidad mínima de rings se estimó en 3, la cantidad media en 6 y la probabilidad de no atender tiene una media del 12 %.

▪ **Transmisión de los datos.**

De acuerdo a la velocidad del módem mencionado anteriormente, y al tamaño de la información a ser transferida, el tiempo dependerá del tipo de comunicación establecida. En el caso de una comunicación máquina-máquina y el tiempo será aproximadamente igual al punto 4. En el caso de una comunicación máquina-hombre el tiempo se incrementará ya que el módem debe detectar que atendió una persona y enviar un mensaje hablado, cuyo volumen de información es mayor que en el caso anterior. Además debe ser un módem con reconocimiento de voz. Se ha estimado una media de 18 segundos en este caso.

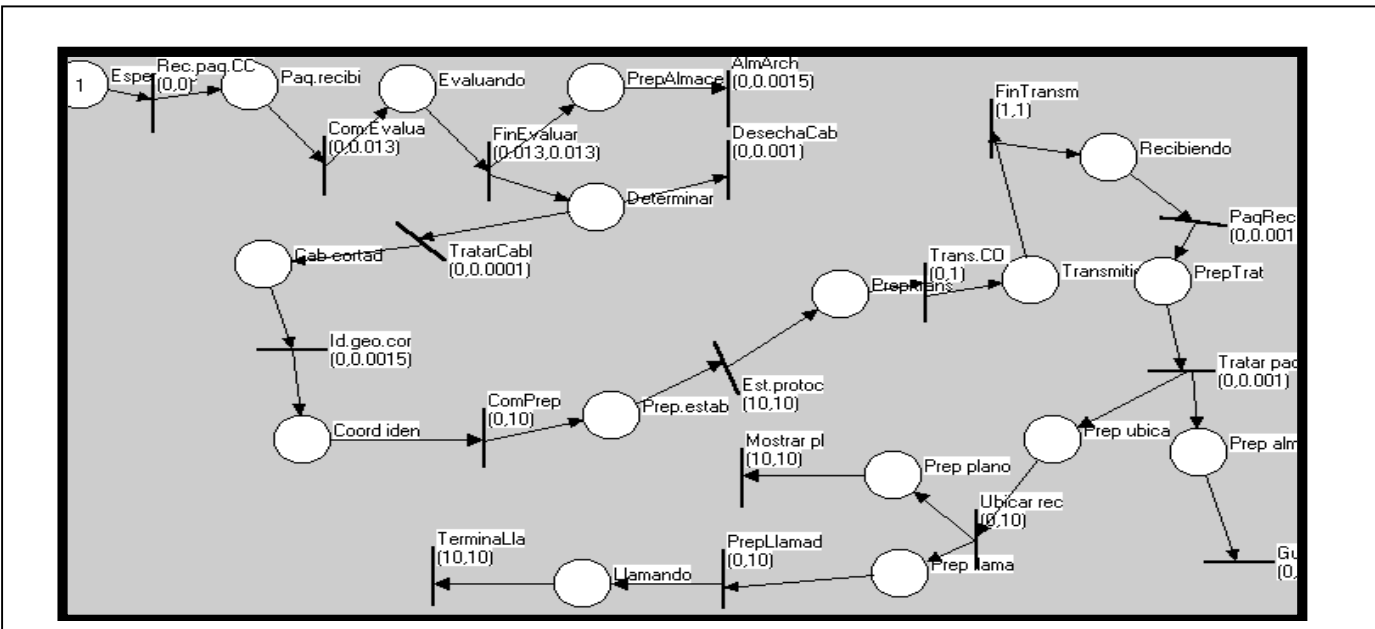
El mejor de los casos sería aquel en el que detectamos un corte y automáticamente los transmitimos al centro de operaciones sin encontrar la línea de comunicación ocupada. El centro de operaciones realiza la ubicación geográfica y se comunica con la comisaría correspondiente a la zona del corte encontrando la línea disponible. El tiempo total aproximado sería de 35 segundos si la comunicación entre el Centro de Operaciones y la comisaría es máquina-máquina, y de 65 segundos aproximadamente si la comunicación entre el Centro de Operaciones y la comisaría es máquina-hombre.

Algunos inconvenientes a tener en cuenta en la modelización de fallas

Todos estos inconvenientes se representaron y estudiaron con la Red de Petri. Estos problemas son:

- a) Al intentar la central comunicarse con el centro de operaciones, éste esté atendiendo otro llamado y la línea se encuentre ocupada.
- b) El centro de operaciones recibió los datos de un cable cortado pero está buscando la información geográfica de un corte anterior, con lo cual se retrasa la búsqueda de los datos de este corte.
- c) El centro de operaciones recibió los datos de un cable cortado y ya buscó la información geográfica del corte, pero el módem está ocupado con la transmisión anterior.
- d) El cable telefónico que une la central y el centro de operaciones está cortado con lo que los cortes no pueden ser reportados.
- e) El cable telefónico que une el centro de operaciones y la comisaría está cortado con lo que los cortes no pueden ser reportados.

El estudio de la modelización de los tiempos se realizó con Redes de Petri extendidas [Roze88] utilizando el ambiente Petri-LIDI [Abas95], [Abas96]. En los gráficos siguientes se muestra la modelización del sistema, directamente tomados desde el ambiente de simulación.

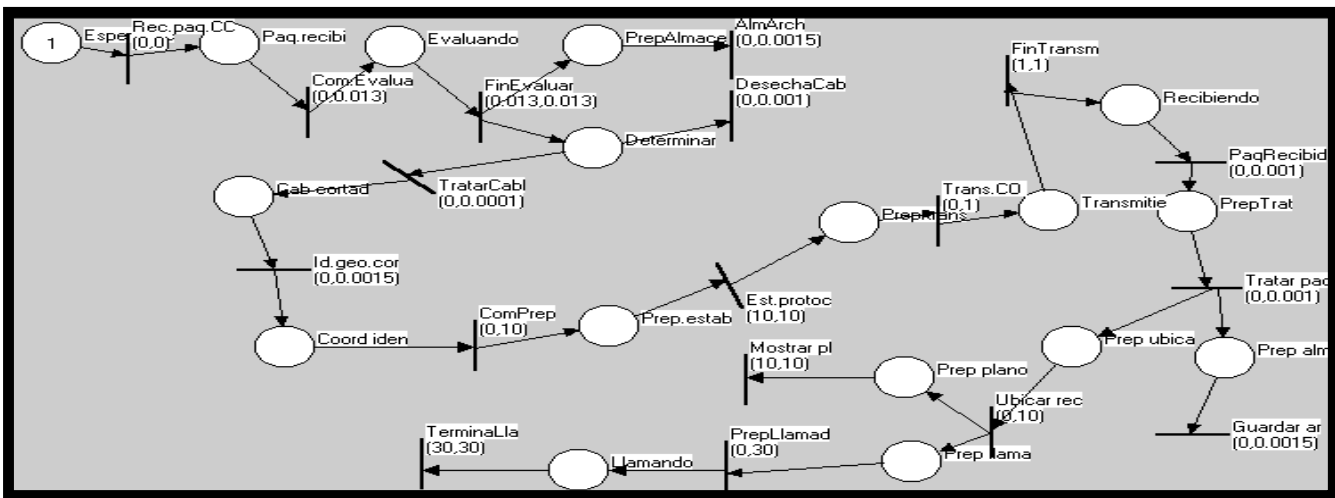


Arriba: Red de Petri con comunicaciones sin reintentos, y máquina-máquina.

Abajo: Red de Petri con comunicaciones sin reintentos, con comunicación máquina-Hombre entre el Centro de Operaciones y la Comisaría.

Toda la documentación referida a la utilización del ambiente, así como los resultados detallados obtenidos están en la bibliografía, pero a modo de síntesis aquí se resumen algunos aspectos resultantes del análisis:

- Se estudió sistemáticamente la posibilidad de 1, 4 y 10 cortes simultáneos, suponiendo las líneas de comunicaciones dedicadas al proceso (sin interferencia) y se obtuvo una media de respuesta del sistema de 37.5 segundos, 49.6 segundos y 84.7 segundos.
- Estudiando los mismos casos, con probabilidad de ocupación en las líneas telefónicas y reintento sistemático: naturalmente los tiempos de respuesta varían



según la probabilidad de ocupación de las líneas y el tiempo pendiente de comunicación. Ajustando estos parámetros se obtuvieron medias de 82.4 segundos, 105.3 segundos y 154.2 segundos. Asimismo incrementando la probabilidad de ocupación y la duración de la misma, se llega a situaciones de imposibilidad de cumplir la restricción total de los 3 minutos. Se infiere que es muy importante para garantizar la respuesta del sistema tener comunicación máquina-máquina y línea dedicada.

- Se buscaron límites de utilidad para la definición del sistema tales como el número máximo de reintentos que hacían posible mantenerse dentro de los 3 minutos indicados en la especificación, tanto para línea ocupada entre el servidor y el puesto local como entre el puesto y la comisaría correspondiente.
- Por último se estudió la mejora del sistema en el caso de poder avisar a una segunda comisaría alternativa en el caso de tener la línea ocupada o no atendida. Esto lleva la probabilidad de aviso seguro dentro de los 3 minutos a 99.1 % suponiendo que no hay falla física en las líneas o modems.

Resultados obtenidos.

El prototipo fue totalmente desarrollado y el ambiente de prueba está operativo en el LIDI (Departamento de Informática, UNLP).

Se utilizó Visual C++ y el sistema de GIS Ageomap 3.5, así como el ambiente Petri-LIDI.

La modelización del hardware de adquisición de datos se hizo en base a los tiempos de muestreo y el protocolo de datos definido en la Facultad de Ingeniería. Del mismo modo se representó la comunicación remota por modem, ya que en las experiencias se utilizaron computadoras dentro de una red local.

El análisis de tiempo deja ciertas garantías de la respuesta en tiempo del sistema real, imponiendo la restricción de tener líneas telefónicas dedicadas. Naturalmente la solución se potencia si se tiene un canal de mayor velocidad o ancho de banda.

Conclusiones y Líneas de Trabajo actuales.

Los problemas de Sistemas de Tiempo Real requieren extensiones a las técnicas clásicas de la Ingeniería de Software. En particular la metodología de Shumate y Keller [Shum92], simplifica la derivación de la implementación de los módulos del prototipo.

El estudio de las respuestas en tiempo, requiere alguna herramienta de ejecución simulada. Las Redes de Petri extendidas [Abas95], demostraron ser útiles y muy flexibles, ya que se pudieron modelar las distintas situaciones planteadas con gran facilidad y claridad, y ejecutarlas obteniendo diferentes tiempos para luego realizar la evaluación de las respuestas del sistema.

En nuestra implementación el manejo de los datos (G.I.S.) se encuentra centralizado en el Centro de Operaciones, con lo cual no ofrece mayores problemas de administración.

Actualmente se trabaja en la integración real del hardware de adquisición de datos (en el prototipo del software, se modeliza el comportamiento del hardware) desarrollado en la Facultad de Ingeniería. Asimismo el producto final de GIS a utilizar puede cambiar (cambiando con ello la respuesta en tiempo y los algoritmos de búsqueda y presentación) con lo que seguramente hay que realizar adaptaciones al prototipo desarrollado.

En cuanto a las extensiones a las Redes de Petri, se pueden incorporar atributos a los tokens y prioridades, lo cual potenciaría la herramienta para algunas clases de problemas. En este caso ha sido suficiente con las prestaciones del ambiente disponible en el LIDI.

Bibliografía

- [Abas95] Abasolo MJ, Cantarella M., De Giusti A. , *"Especificacion y Simulacion de Sistemas de Tiempo Real con Redes de Petri extendidas"*, Proceedings CRITEC 95. Brasil 1995.
- [Abas96] Abasolo MJ, Cantarella M., De Giusti A., Naiouf M. *"Petri-LIDI: Un ambiente de verificación de restricciones de tiempo en Sistemas de Tiempo Real"*, Informe Técnico LIDI-1996.
- [Burn96] Burns A., Wllings A., *"Real-Time Systems and Programming Languages"*, Addison Wesley, 1996.
- [Davi93] Davis Alan, *"Software Requirements, Objects, Functions & States"*, Prentice Hall, 1993.
- [Fern98] Fernandez P., Artime A., De Giusti A., Ramon H., *"Sistema distribuído de Tiempo Real para la detección de cortes en la Red telefónica"*, Tesina de Licenciatura en Informática. UNLP. 1998.
- [Geha86] N. Gehani, A. D. McGettrick, *"Software Specification Techniques"*, Addison Wesley, 1986.
- [Härb94] Härbour M., Klein M., Lehoczky J. *"Timing Analysis for Fixed-Priority Scheduling of Hard Real-Time Systems"*, IEEE Transactions on Software Engineering, Enero 1994, VOL 20, pag. 13.
- [Josep96] Joseph M, *"Real-Time Systems: Specification, verification and Analysis"*, Prentice-Hall, 1996.
- [Lapl98] Laplante Phillip, *"Design and application of real time systems"*, IEEE Press, 1998.
- [Levi90] Shem-Tov Levi, Ashok Agrawala, *"Real Time System Design"*, McGraw-Hill Inc, 1990.
- [Niel90] Kjell Nielsen, *"Ada in Distributed Real-Time Systems"*, McGraw-Hill, 1990.
- [Pete81] J. Peterson, *"Petri Nets Theory and the Modeling of Systems"*, Prentice-Hall, 1981.
- [Roze88] Rozenberg (Ed.), *"Advances in Petri Nets"*, Springer-Verlag, 1988.
- [Sand89] Sanden B., *"Entity-Life modeling and structured analysis in Real Time software design. A comparison"*. Communication of the ACM. Vol 32, Número. 12, Dec. 1989. pp 1458-66.
- [Shum92] Shumate K., Keller M., *"Software specification and design. a disciplined approach for real-time systems"*, Wiley 1992.
- [Ward85] Ward P., Mellor S., *"Structured development for real time systems"*, Vol 1-3, Yourdon Press, 1985.