

Verificación de las restricciones de tiempo.

Lic. Patricia M. Pesado¹, Ing. Armando E. De Giusti²

*Laboratorio de Investigación y Desarrollo en Informática³
Departamento de Informática - Facultad de Ciencias Exactas
Universidad Nacional de La Plata*

Descripción

La naturaleza de los sistemas distribuidos de tiempo real (SDTR) requiere técnicas de análisis, diseño y prueba específicas que permitan representar las situaciones típicas de estos sistemas.

Un sistema distribuido de tiempo real debe interactuar con el mundo real, en puntos físicamente distantes, en períodos de tiempo que vienen determinados por el contexto o las restricciones de la especificación (en muchos casos a partir de una activación asincrónica).

La evolución tecnológica en el tratamiento de señales (locales o remotas) y en los sistemas de comunicaciones ha impulsado enormemente esta área temática, sobre todo en los aspectos de planificación y desarrollo de software para Sistemas Distribuidos de Tiempo Real [HAT88][LAP93][SHU92]

Como cualquier sistema basado en computadora, un SDTR debe integrar software, hardware, personas, bases de datos, no solo para cumplir con los requisitos funcionales del sistema, sino también con los requerimientos de rendimiento.

Las dificultades principales del desarrollo de software para sistemas de tiempo real son:

- Controlar hardware en forma directa
- Procesar mensajes que arriban en forma asincrónica, con diferentes velocidades y diferentes prioridades
- Detectar y controlar condiciones de falla. Prever diferentes grados de recuperación del sistema.
- Manejar colas y buffers de almacenamiento y mensajes.
- Modelizar condiciones de concurrencia en un conjunto apropiado de procesos.
- Asignar procesos lógicos a procesadores físicos (si se dispone de ellos).
- Manejar las comunicaciones inter-procesos e inter-procesadores.
- Proteger datos compartidos por procesos concurrentes.
- Organizar (schedule) y despachar la atención de procesos.
- Manejar las restricciones de tiempo y performance.
- Relacionarse con un reloj de tiempo real y evitar desfasajes de procesadores – tiempos.
- Testear y poner a punto un sistema que normalmente esta distribuido en diferentes procesadores.
- Elaborar herramientas de software que permitan simular o emular dispositivos o eventos de hardware no disponibles en el desarrollo.
- Reducir y estructurar los requerimientos.
- Seleccionar la estructura de hardware adecuada.

Cada una de las actividades del ciclo de vida de los sistemas necesita de técnicas propias o extendidas para hacer frente a la descripción de este tipo de sistemas:

- En la etapa de planificación es importante contar con estimaciones fiables que permitan proyectar efectivamente el proceso de desarrollo. La recolección de métricas que conformen una línea base para la gestión de proyectos de características comunes.
- En la etapa de análisis deben lograrse especificaciones que permitan representar los flujos y procesamientos de control, los flujos de información que son producidos o recogidos continuamente, las ocurrencias múltiples de la misma transformación, los estados del sistema y mecanismos que producen las transiciones.
- En la etapa de diseño debe tenerse en cuenta la coordinación entre las tareas, el procesamiento de interrupciones, el manejo de E/S (que asegura la no pérdida de datos), la especificación de las ligaduras de tiempo externas, el aseguramiento de la consistencia de su base de datos.
- En la etapa de pruebas del sistema deben diseñarse casos de prueba para el tratamiento de eventos (ej: proceso de interrupciones), la temporización de los datos, el paralelismo de las tareas. La estrategia de pruebas puede implementarse a través de la prueba de tareas, de comportamiento, intertareas y del sistema.
- Las herramientas CASE permiten crear modelizaciones para simular el comportamiento de los sistemas de tiempo real como consecuencia de sucesos externos, debido a que no siempre es posible probar sobre el sistema.

Objetivos de la línea de investigación

En el marco de los proyectos de investigación del LIDI sobre Sistemas de Tiempo Real y Sistemas Distribuidos en Automatización de Oficinas se han realizado varias experiencias de desarrollo de Sistemas Distribuidos de Tiempo Real para áreas de aplicación muy diferentes.

- Desde el punto de vista de las especificaciones se han desarrollado herramientas de derivación automática de código, en particular especificaciones en lenguaje Estelle y CSP derivadas a código ADA ejecutable. También desde Redes de Petri extendidas se ha derivado prototipos de tareas en ADA. Se han aplicado estos trabajos especialmente a la concepción y validación de protocolos.
- Dado que el desarrollo de sistemas de tiempo real implica la aplicación de los conceptos de Ingeniería de Sistemas, se han analizado las extensiones a las metodologías estructuradas propuestas por Keller y Shumate y posteriormente se aplicaron las mismas a la modelización y desarrollo de la Ingeniería de Software de un sistema de supervisión y control de cables telefónicos, que debe detectar cortes en tiempo real, incluyendo la ubicación de los mismos y el aviso a la unidad policial correspondiente en menos de 3'.

Cada una de las actividades del ciclo de vida de los sistemas necesita de técnicas propias o extendidas para hacer frente a la descripción de este tipo de sistemas:

- En la etapa de planificación es importante contar con estimaciones fiables que permitan proyectar efectivamente el proceso de desarrollo. La recolección de métricas que conformen una línea base para la gestión de proyectos de características comunes.
- En la etapa de análisis deben lograrse especificaciones que permitan representar los flujos y procesamientos de control, los flujos de información que son producidos o recogidos continuamente, las ocurrencias múltiples de la misma transformación, los estados del sistema y mecanismos que producen las transiciones.
- En la etapa de diseño debe tenerse en cuenta la coordinación entre las tareas, el procesamiento de interrupciones, el manejo de E/S (que asegura la no pérdida de datos), la especificación de las ligaduras de tiempo externas, el aseguramiento de la consistencia de su base de datos.
- En la etapa de pruebas del sistema deben diseñarse casos de prueba para el tratamiento de eventos (ej: proceso de interrupciones), la temporización de los datos, el paralelismo de las tareas. La estrategia de pruebas puede implementarse a través de la prueba de tareas, de comportamiento, intertareas y del sistema.
- Las herramientas CASE permiten crear modelizaciones para simular el comportamiento de los sistemas de tiempo real como consecuencia de sucesos externos, debido a que no siempre es posible probar sobre el sistema.

Objetivos de la línea de investigación

En el marco de los proyectos de investigación del LIDI sobre Sistemas de Tiempo Real y Sistemas Distribuidos en Automatización de Oficinas se han realizado varias experiencias de desarrollo de Sistemas Distribuidos de Tiempo Real para áreas de aplicación muy diferentes.

- Desde el punto de vista de las especificaciones se han desarrollado herramientas de derivación automática de código, en particular especificaciones en lenguaje Estelle y CSP derivadas a código ADA ejecutable. También desde Redes de Petri extendidas se ha derivado prototipos de tareas en ADA. Se han aplicado estos trabajos especialmente a la concepción y validación de protocolos.
- Dado que el desarrollo de sistemas de tiempo real implica la aplicación de los conceptos de Ingeniería de Sistemas, se han analizado las extensiones a las metodologías estructuradas propuestas por Keller y Shumate y posteriormente se aplicaron las mismas a la modelización y desarrollo de la Ingeniería de Software de un sistema de supervisión y control de cables telefónicos, que debe detectar cortes en tiempo real, incluyendo la ubicación de los mismos y el aviso a la unidad policial correspondiente en menos de 3'.

- La necesidad de contar con estimaciones tempranas para sistemas de tiempo real ha impulsado al estudio de las métricas de Punto Función extendidas. Resultados de estas investigaciones se aplicaron a la evaluación de un sistema distribuido de comunicaciones utilizado en entrenamiento militar.
- La complejidad de los sistemas a modelizar hace necesaria la utilización de herramientas CASE . Se ha investigado en la especificación de herramientas CASE orientadas a tiempo real, incluyendo desarrollos experimentales.
- La necesidad de poder especificar y verificar restricciones de tiempo hace necesario incorporar a las herramientas de análisis y diseño, la posibilidad de modelar las especificaciones temporales y poder verificarlas mediante simulación. Para esto se desarrollaron dos ambientes experimentales centrados en Redes de Petri extendidas.

Mecanismos de especificación y verificación de restricciones de tiempo

Existen numerosos enfoques que tienden a la especificación y verificación de restricciones de tiempo entre los que se cuentan los lenguajes de simulación, las aproximaciones axiomáticas y las Redes de Petri.

En particular las Redes de Petri son un elemento útil para modelar sistemas en los que existen dependencias secuenciales dinámicas y consideraciones de procesamiento concurrente, por las siguientes características:

- Son inherentemente paralelas y soportan eventos asincrónicos.
- Permiten una representación explícita de las dependencias causales y de las interdependencias entre procesos.
- Permiten la descripción de un sistema en diferentes niveles de abstracción
- Favorecen la verificación y en particular la detección de situaciones críticas (por ejemplo deadlocks).
- Soportan extensiones para manipular temporizaciones aleatorias y/o determinísticas.

Para poder utilizarlas en modelizaciones de Sistemas de Tiempo Real se incorporan restricciones de tiempo y se han extendido las clases de arcos del grafo de Petri para simular condiciones de funcionamiento. De esta manera pueden modelarse fenómenos tales como tiempo entre llegadas de eventos externos, tiempos de reacción, duración de actividades, tiempos de timeout, acciones periódicas, etc.

Actualmente se está investigando la incorporación de las Redes de Petri extendidas a ambientes CASE orientados a tratamiento de sistemas de tiempo real y asimismo se están estudiando variantes (en particular Queued Petri Nets y la incorporación de atributos a los tokens) para analizar mejor fenómenos de espera y de manejo de datos distinguidos dentro del sistema.

Bibliografía

- [Aba95]** Abasolo, Cantarella, De Giusti "Especificación y Simulación de Sistemas de Tiempo Real con Redes de Petri Extendidas (RPE) - Derivación de Código a partir de una RPE ". Proceedings CRICTE 95 – UFRGDS – Brasil.
- [And91]** Andrews, "Concurrent Programming", Benjamin/Cummings, 1991.
- [Geh86]** N. Gehani, A. D. McGettrick, "Software Specification Techniques", Addison Wesley, 1986.
- [Hat88]** Hatley D., Pirbhai I., "Strategies for Real-Time System Specification", Dorset House, 1988.
- [Hoa85]** Hoare C, "Communicating Sequential Processes", Englewood Cliffs, Prentice-Hall, 1985.
- [Kar92]** Karonis N, "Timing Parallel Programs That Use Message Passing", Journal of Parallel and Distributed Computing, 14, 1992.
- [Lap88]** Laplante P. "Design and Applications of Real Time Systems", IEEE Press 1988.
- [Law92]** H. Lawson, "Parallel processing in industrial real time applications", Prentice Hall 1992.
- [Lev90]** Shem-Tov LEVI, Ashok Agrawala, "Real Time System Design", McGraw-Hill Inc, 1990.
- [Nie90]** Kjell Nielsen, "Ada in Distributed Real-Time Systems", McGraw-Hill, 1990.
- [Pet81]** J. Peterson, "Petri Nets Theory and the Modeling of Systems", Prentice-Hall, 1981.
- [Pre98]** Pressman R., "Ingeniería de Software", McGraw Hill, 1992.
- [Rom98]** Romero F., Sosa R. F., De Giusti A , "Sistemas Distribuidos de Tiempo Real. Experiencia en el diseño y puesta a punto de un sistema de mensajería para entrenamiento militar" . Proceedings CACIC 98.
- [Roz88]** Rozenberg (Ed.), "Advances in Petri Nets", Springer-Verlag, 1988.
- [Shu92]** Shumate K., "Software specification and design for real-time systems", Wiley 1992.

¹ Profesor Titular, Ded. Exclusiva, Dpto. de Informática, Facultad de Cs. Exactas, UNLP.
Profesional de la CIC. E-mail: ppesado@info.unlp.edu.ar

² Inv. Principal CONICET. Profesor Titular Ded. Excl., Dpto. de Informática, Facultad de Cs. Exactas, UNLP. E-mail: degiusti@info.unlp.edu.ar

³ Calle 50 y 115 Primer Piso, (1900) La Plata, Argentina, Teléfono 54-221-4227707
WEB: lidi.info.unlp.edu.ar