

INTEGRACION DE MODELOS DE DESEMPEÑO EN UNA METODOLOGIA DE DESARROLLO DE SISTEMAS

Alberto VALDERRUTEN - Yves RAYNAUD

Institut de Recherche en Informatique de Toulouse
Equipe SIERA - Université Paul Sabatier
118, Route de Narbonne 31062 Toulouse Cedex - France
e-mail: valderru@irit.fr

Resumen

Cuando la especificación de un sistema incluye restricciones de desempeño como los tiempos de respuesta esperados, su diseño debe realizarse siguiendo criterios cuantitativos adicionales, que se toman en cuenta mediante Modelos de Desempeño.

En este artículo situamos la integración del proceso de Evaluación del Desempeño en el desarrollo de un sistema, vista en términos de coexistencia entre Modelos de Diseño. Los Modelos de Desempeño deben reflejar la información contenida en otros modelos; con este propósito, identificamos las interacciones entre la Metodología de Desarrollo del sistema y el proceso de Evaluación del Desempeño.

Como ilustración, derivamos un Modelo de Redes de Colas a partir de un modelo construido con la Técnica de Descripción Formal LOTOS. Este trabajo fue realizado en el marco del proyecto COMPLEMENT, cuyo propósito es el estudio, adecuación e integración de métodos y herramientas para el desarrollo de Sistemas Tiempo Real.*

Palabras clave: Modelos de Desempeño, Modelos de Diseño, Proceso de Evaluación del Desempeño, Metodología de Desarrollo de Sistemas, Técnicas de Descripción Formal, Redes de Colas.

* *COMPLEMENT* es un proyecto financiado parcialmente por la CEE en el programa ESPRIT II (proyecto No. 5409).

1. Objetivos

Las técnicas de Evaluación del Desempeño pueden darnos información sobre el comportamiento de un sistema visto en términos de tiempo, y así estimar la calidad del servicio ofrecido (tiempos de respuesta) al igual que su eficacia (utilización correcta de los recursos). Estas técnicas pueden clasificarse según utilicen o no alguna forma de abstracción (modelos) que permitan deducir los valores para los diferentes Criterios de Desempeño. Entre las técnicas de modelado para la Evaluación del Desempeño más populares, encontramos las Redes de Colas [4, 5, 7], que permiten expresar los fenómenos de contención que se presentan en un Sistema Informático.

Realizar una Evaluación del Desempeño al mismo tiempo que el sistema es desarrollado implica tener en cuenta su comportamiento como criterio de diseño, lo cual es determinante en aquellos sistemas de reacción rápida (Tiempo Real, Sistemas Embarcados) y cada vez más exigido en sistemas de tipo conversacional. Sin embargo, los métodos de desarrollo tradicionales no permiten la expresión de exigencias de desempeño en términos de tiempo, ya que enfocan principalmente los “aspectos funcionales” (el sistema debe realizar una serie de funciones) dejando a un segundo plano los “aspectos no funcionales”, que son tomados en cuenta una vez el sistema realizado. Los inconvenientes de esta práctica que son ampliamente comentados en [10].

Nuestro objetivo es incluir la Evaluación del Desempeño en el desarrollo del sistema, apoyándonos en técnicas de modelado, lo que supone que los modelos obtenidos deben integrarse en la metodología de desarrollo utilizada. Un análisis del papel que pueden cumplir estos modelos en las diferentes etapas del ciclo de vida del sistema fue ya presentado en [1]. El siguiente paso es estudiar la complementaridad entre los Modelos de Desempeño que pretendemos introducir y los que se utilizan para diseñar el sistema en el marco de una metodología de desarrollo.

2. Integración de Modelos

El Diseño de un sistema se realiza generalmente mediante un proceso iterativo: una hipótesis es formulada, representada con la ayuda de algún formalismo (es decir, modelada), evaluada según criterios de eficacia y de viabilidad, y si la evaluación no satisface entonces se revisa la hipótesis.

Los estudios de modelado para la Evaluación del Desempeño siguen un proceso análogo: una técnica de modelado del desempeño es el formalismo de representación (las Redes de Colas, por ejemplo), y los criterios de evaluación adicionales son las exigencias cuantitativas de desempeño.

Un conjunto de modelos puede así coexistir dentro de un mismo proceso de diseño, cada uno de ellos poniendo de relieve un aspecto específico del sistema. Una relación existe entre el grado de abstracción de cada modelo, la técnica que lo engendra y los criterios de evaluación que lo motivan. El método de desarrollo se define como el conjunto de actividades que conduce a la obtención de

uno o varios modelos utilizando una o varias técnicas. Lo que nos proponemos es estudiar la manera de integrar una nueva técnica (en este caso, de modelado del desempeño) a un método de desarrollo.

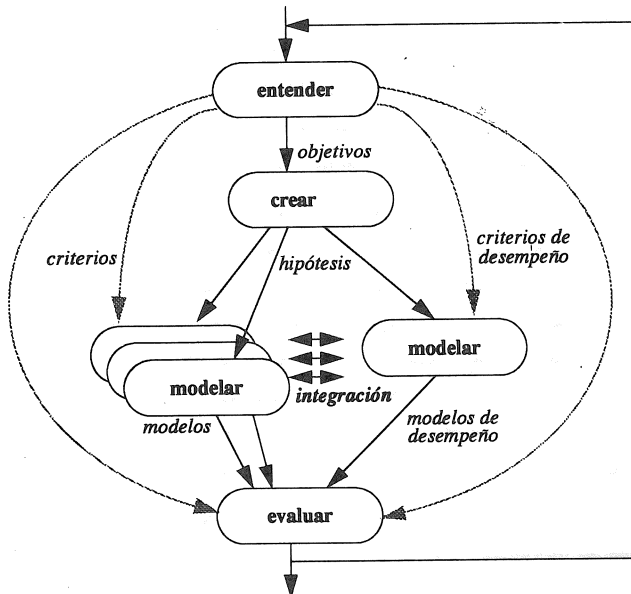


Fig. 1: Modelos en el Proceso de Diseño

La figura 1 sitúa la problemática de integración de un proceso de Evaluación del Desempeño en términos de coexistencia entre varios modelos. Aunque cada uno de ellos presente el diseño del sistema bajo un punto de vista diferente, ciertos conceptos se pueden encontrar repetidos en varios modelos. La idea es facilitar la creación de Modelos de Desempeño, utilizando al máximo la información procedente de otros modelos.

3. Puntos de Interacción con el Desarrollo del Sistema

El estudio de las interacciones entre el proceso de modelado del Desempeño y el proceso de desarrollo del sistema nos conduce a la definición de un conjunto de puntos de interacción. Cada uno está definido mediante la información que se intercambia en ellos: podemos detectar así el instante apropiado para emprender un estudio de Evaluación del Desempeño durante el desarrollo del sistema.

La interacción entre el proceso de Evaluación del Desempeño y el Desarrollo del Sistema tiene lugar básicamente en cinco puntos:

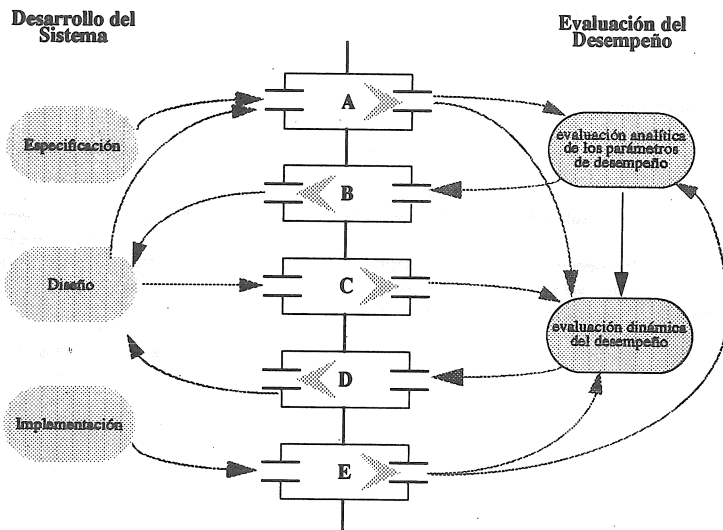


Fig. 2: Puntos de Interacción con el Desarrollo del Sistema

Los puntos A, C y E son entradas para el proceso de Evaluación del Desempeño. La información correspondiente está constituida por:

- las exigencias de desempeño en términos de tiempo
- las hipótesis de diseño consideradas
- un conjunto de factores de comportamiento, que pueden ser medidos sobre otros sistemas, calculados a partir de las características conocidas, o sencillamente estimados

Los puntos B y D corresponden a los resultados del proceso de Evaluación del Desempeño, que incluyen:

- las estimaciones obtenidas a partir de los modelos sobre el comportamiento general del sistema
- las alternativas de diseño que se puedan identificar en el caso de que algún problema en el comportamiento fuera detectado

La Tabla 1 detalla la información implicada en los puntos de interacción.

4. El proceso de Evaluación del Desempeño

Un modelo del proceso de Evaluación del Desempeño fue presentado en [3]. Su contenido no se detallará en este artículo. Sin embargo, al estudiar su interacción con el desarrollo del sistema se destacan dos etapas de este proceso (Fig. 2):

- una evaluación analítica de los parámetros de desempeño, necesaria para la construcción de los modelos [10], que además permite realizar un análisis "optimista" del comportamiento del sistema,

donde los problemas de contención no son aún tomados en cuenta;

- una evaluación dinámica del comportamiento del sistema, resultado de la utilización de los Modelos de Desempeño.

El resultado de este proceso es la evaluación del Diseño con los criterios de desempeño.

entrada A	salida B	entrada C	salida D	entrada E
exigencias de desempeño (medidas especificadas)	frecuencias de solicitud de los componentes	configuración de los recursos (topología)	<i>throughput</i> (cantidad de trabajo útil)	escenarios representativos de la carga
escenarios representativos de la carga	tiempos de servicio de los componentes	parámetros de configuración (talla de memorias...)	utilización de los componentes	diagramas de ejecución (incluyen probabilidades)
talla de los componentes del <i>software</i>	consumo de recursos / componente	parámetros de carga complementarios	tiempos de respuesta	factores de desempeño del <i>hardware</i>
diagramas de ejecución (incluyen probabilidades)	consumo total de recursos / escenario	políticas de gestión de los recursos	ocupación media del sistema	utilización de los recursos por los componentes
factores de desempeño del <i>hardware</i>	tiempos de respuesta "optimistas"	factores de gestión (<i>overheads</i>)	tiempos de espera / tiempos de acceso	parámetros de carga complementarios
factores de desempeño del entorno operativo	utilización media de los recursos <i>hardware</i>	alternativas de diseño	capacidad del sistema	políticas de gestión de los recursos
utilización de los recursos por los componentes	alternativas de diseño		recursos críticos (<i>bottlenecks</i>)	factores de gestión (<i>overheads</i>)
			alternativas de diseño / configuración	alternativas de configuración

Tabla 1: Información de desempeño

Frente a las diferentes etapas del desarrollo del sistema, identificamos qué tipo de información debe utilizarse en el modelado del desempeño. Esta información proviene de otros modelos de desarrollo.

De las Especificaciones se extraen (entrada A) las restricciones de desempeño en términos de tiempo (generalmente se especifican los tiempos de respuesta esperados). Estas exigencias sólo pueden ser consideradas de manera efectiva si son lo suficientemente precisas [12, 13], y están acompañadas de escenarios de utilización del sistema, lo que supone un trabajo importante a nivel del análisis de las necesidades.

Del Diseño extraemos (entradas A y C) la información correspondiente a las estructuras de

ejecución de los escenarios, las características del ambiente de implantación considerado, la utilización de sus recursos y las alternativas de diseño que se quieran comparar.

Desde las primeras etapas de la Implementación podemos obtener una serie de valores medidos sobre el sistema (entrada E), que reemplazan a las estimaciones hechas para la construcción de los modelos. El resultado es una herramienta para la validación del sistema, al mismo tiempo que se mejoran las estimaciones para proyectos futuros. Estas ventajas dependen de un esfuerzo de instrumentación del sistema con monitores, cuyo costo disminuye sensiblemente si su realización está prevista durante el Diseño del sistema. No nos extenderemos más con otras aplicaciones que pueden tener los monitores, pero evidentemente la validación de Modelos de Desempeño no es la única actividad favorecida con su implementación.

5. Aplicación

El estudio de integración de modelos que presentamos a continuación fue realizado en el marco del proyecto COMPLEMENT, en el cual se realiza un panorama de técnicas, métodos y herramientas para el desarrollo de Sistemas Tiempo Real, con el objetivo de proponer soluciones de integración completas y viables.

Presentamos un ejemplo de derivación de un Modelo de Desempeño a partir de un modelo de diseño. Para evitar ambigüedades en la descripción de los modelos de diseño, preferimos partir de una Técnica de Descripción Formal estándar como LOTOS [14]. El uso de este tipo de técnicas, en constante progresión, se justifica por sus capacidades de Verificación y Validación. Al integrar el modelado del desempeño, pretendemos añadir otra dimensión de verificación, guiada por la predicción del comportamiento.

Escogimos un problema suficientemente analizado como para tener una buena referencia para la validación de nuestros modelos: un protocolo de comunicación tipo *stop & wait*. La información es transmitida en un sólo sentido, en paquetes, de un emisor a un receptor. Se supone un canal de comunicación que puede alterar o perder los paquetes en ambos sentidos, así que el emisor espera la confirmación que el receptor envía al recibir un paquete correcto antes de emitir un nuevo paquete. En caso de error, el receptor no responde. El emisor gestiona con un mecanismo de *time-out* la reemisión de los paquetes erróneos o perdidos.

El modelo de comportamiento del protocolo especificado con LOTOS se presenta en la figura 3. En él, dos procesos paralelos *transmitter* y *receiver* se sincronizan con un proceso *medium* a través de dos puntos de sincronización *send* y *rec*. La alteración de paquetes se asimila a su pérdida en el *medium*. El operador \square expresa el indeterminismo en el comportamiento de un proceso. Para una descripción completa de la semántica de LOTOS, nos referimos a [14].

```

specification stop&wait [get, give]: noexit
behaviour
hide send, rec in
  (transmitter [get, send, rec] ||| receiver [give, send, rec])
  |[send, rec]|
  medium [send, rec])
where
process transmitter [get, send, rec]: noexit :=
  get; send!info; sending [get, send, rec]
where
process sending [get, send, rec]: noexit :=
  rec !ack; transmitter [get, send, rec]
  [] i; send !info; sending [get, send, rec] (* timeout *)
endproc (* sending *)
endproc (* transmitter *)
process receiver [give, send, rec]: noexit :=
  rec !info; give; send!ack; receiver [give, send, rec]
endproc (* receiver *)
process medium [send, rec]: noexit :=
  send!info;
  (
    ( i ; rec!info; medium [send, rec])
    [] ( i ; medium [send, rec]))
[] send!ack;
  (
    ( i ; rec !ack; medium [send, rec])
    [] ( i ; medium [send, rec]))
endproc (* medium *)
endspec (* stop&wait *)

```

Fig. 3: Modelo LOTOS del protocolo stop & wait

El objetivo de una Evaluación del Desempeño es en este caso la predicción del *Throughput* del canal de comunicación. Rico y Bochmann presentan los resultados obtenidos con diferentes técnicas de modelado del desempeño [9], y los comparan a los que obtienen al utilizar una variante de LOTOS (Stochastic LOTOS) que introduce el factor tiempo. A diferencia de este procedimiento que modifica el método de diseño, en este caso LOTOS, optamos por hacer uso de su misma semántica para derivar un Modelo de Desempeño, utilizando las Redes de Colas como técnica y el lenguaje de modelado QNAP2 [8] como herramienta.

El modelo de Redes de Colas del protocolo *stop & wait* de la figura 4 se obtuvo a partir de su modelo LOTOS y de las mismas especificaciones de desempeño utilizadas en [6, 9]:

- escenario: el emisor siempre tiene paquetes por enviar al receptor
- el canal de comunicación es de 9600 bps
- paquetes de 1024 bits (en ambos sentidos), tratados en cada extremidad durante 13.5 ms
- el *overhead* de emisión es de 1 ms
- la probabilidad de pérdida de un paquete es del 5%
- *timeout* de 1 s para la reemisión de un paquete

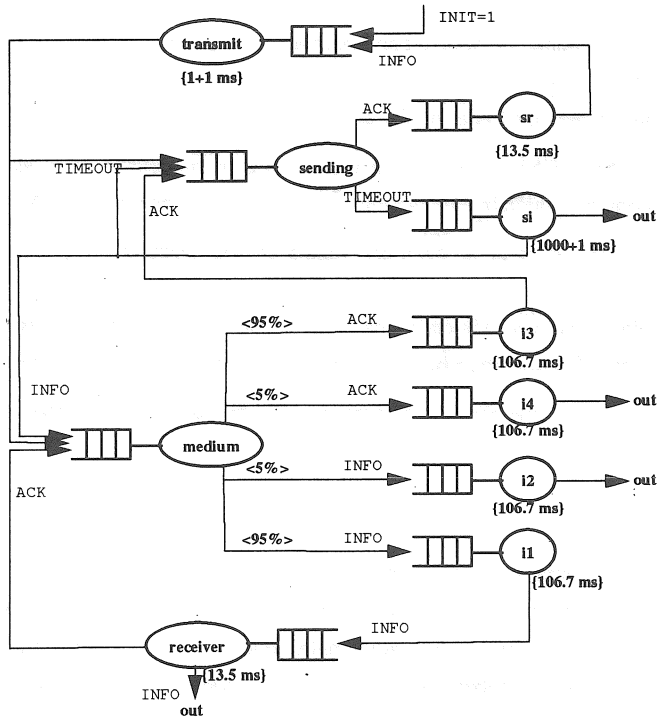


Fig. 4: Modelo de Redes de Colas del protocolo stop & wait

La simulación de este modelo produjo resultados muy próximos a los de referencia (Tabla 2). Cabe señalar que no se aplicaron los métodos analíticos de Redes de Colas (exactos o aproximados) para la resolución del modelo, ya que los mecanismos de modelado utilizados no están cubiertos por la teoría de Redes de Colas.

distribución de los tiempos de servicio	resultados obtenidos con el modelo derivado	resultado de [9] con Stochastic LOTOS	resultado de [6] con Stochastic Petri Nets
Uniforme	2.855	2.852	
Exponencial	2.863		2.75

Tabla 2: Throughput en paquetes por segundo

Con la construcción de este Modelo de Desempeño deducimos una primera serie de reglas de transformación de modelos LOTOS en modelos de Redes de Colas [2, 11]. El esfuerzo de modelado del desempeño puede verse reducido considerablemente, y limitarse a la introducción de la información relativa a las especificaciones de desempeño en un modelo de diseño.

Para tomar en cuenta estas especificaciones al mismo tiempo que se construye el modelo LOTOS, proponemos comentar el modelo con información de desempeño que se sitúe al mismo nivel de abstracción. La derivación del Modelo de Desempeño incluye la transformación de los comentarios en parámetros para las Redes de Colas.

Identificamos dónde deben situarse los comentarios en el modelo LOTOS, así como qué información deben contener; inicialmente, adoptamos la solución consistente en añadir:

- los tiempos de ejecución de las acciones: cada acción se acompaña de $\{t\}$, el tiempo necesario para su ejecución.
- la probabilidad relativa a cada encadenamiento de acciones: el indeterminismo se expresa con la notación $\langle w_1 \rangle B_1 [] \dots [] \langle w_n \rangle B_n$, donde w_i es la frecuencia relativa del comportamiento B_i .

Esta información adicional permitió parametrizar el modelo de Redes de Colas con los valores que aparecen en la figura 4.

6. Conclusión

El desarrollo de sistemas de reacción rápida (Tiempo Real, Sistemas Embarcados) o de tipo conversacional que incluyan fuertes exigencias en términos de tiempo (tiempos de respuesta) debe realizarse siguiendo criterios cuantitativos adicionales, que se toman en cuenta mediante modelos de Desempeño. Estos modelos deben coexistir con otros modelos de diseño, bien sea porque no cubren la totalidad de los aspectos funcionales del diseño, o porque otros criterios no funcionales (seguridad, robustez...) se consideran igualmente importantes.

Teniendo en cuenta este hecho, identificamos una serie de puntos de interacción entre el proceso de desarrollo del sistema (y sus modelos de diseño) y el de Evaluación del Desempeño (y sus modelos de comportamiento). Nos interesamos en la información necesaria para construir Modelos de Desempeño, identificando aquella que podemos extraer de otros modelos de diseño.

El proceso de modelado para la Evaluación del Desempeño puede facilitarse si se crean los mecanismos necesarios a la reutilización de los aspectos funcionales del diseño que aparecen en los modelos normalmente utilizados para el desarrollo del sistema.

Estudiamos la posibilidad de derivar un modelo de Redes de Colas a partir de un modelo realizado con el formalismo de LOTOS, en el cual añadimos la información sobre los tiempos de ejecución de las acciones y sus probabilidades de encadenamiento como simples comentarios que son utilizados para la derivación. Aplicamos este proceso en el caso del protocolo *stop & wait*, validamos el modelo de Redes de Colas obtenido, y deducimos un primer conjunto de reglas de transformación.

Estas reglas deberán permitir la automatización del proceso. De esta manera, el esfuerzo de modelado del desempeño se concentra en la introducción de los factores de comportamiento, que pueden ser medidos, calculados o estimados. Pero antes de llegar a una derivación automática de modelos de Redes de Colas a partir de LOTOS, dos actividades tienen que ser completadas:

- Validar el conjunto de reglas de derivación con el estudio de otro tipo de aplicaciones, verificando que se cubren los mecanismos de modelado más comunes;
- Estudiar la relación entre los factores de comportamiento, con el fin de establecer el conjunto mínimo de información adicional que debe incluirse en los modelos de diseño para cubrir las exigencias de desempeño.

7. Bibliografía

- [1] J. Aguilar, A. Valderruten, *La Evaluación de Desempeño en una Metodología de Desarrollo de Sistemas de Información Distribuidos*, XVII Conf. Latinoamericana PANEL91, Caracas, Julio 8-12, 1991.
- [2] A. Benzekri, D. Gazal, O. Hjej, A. Valderruten, *LOTOS / Queueing Networks integration*, Proyecto COMPLEMENT, internal report IRI29-1.0, Enero 13 de 1992.
- [3] E. Conquet, A. Valderruten, R. Trémoulet, Y. Raynaud, S. Ayache, *A Process Model of the Performance Evaluation Activity*, Summer Comp. Simulation Conf., Reno, Nevada, Julio 27-30, 1992.
- [4] L. Kleinrock, *Queueing Systems*, Wiley-Interscience, 1975.
- [5] S. S. Lavenberg, *Computer Performance Modeling Handbook*, Academic Press, 1983.
- [6] M.K. Molloy, *Performance Analysis Using Stochastic Petri Nets*, IEEE Trans. on Computers, Vol. C-31, No. 9, Septiembre 1982.
- [7] G. Pujolle, S. Fdida, *Modèles de Systèmes et de Réseaux, II - Files d'Attente*, Eyrolles, 1989.
- [8] QNAP2, *Reference Manual, V4.0*, Simulog, Bull e INRIA, 1984.
- [9] N. Rico, G. v. Bochmann, *Performance description and analysis for distributed systems using a variant of LOTOS*, Tenth International IFIP WG6.1 Symp. on Protocol Specification, Testing and Verification, Julio 12-15, 1990.
- [10] C. U. Smith, *Performance Engineering of Software Systems*, Addison-Wesley, 1990.
- [11] A. Valderruten, O. Hjej, A. Benzekri, D. Gazal, *Deriving Queueing Networks Performance Models from Annotated LOTOS Specifications*, 6th Int. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation, Edinburgo, Escocia, Septiembre 16-18, 1992.
- [12] ESA PSS-05-0 Issue 2, *ESA Software Engineering Standards*, Febrero 1991.
- [13] IEEE 610.12-1990, *IEEE Standard Glossary of Software Engineering Terminology*, Diciembre 10, 1990.
- [14] ISO/TC97/SC21, tech. report No. IS 8807, *LOTOS: a formal description technique based on the temporal ordering of observational behaviour*, 1987.