

# Simulación del Modelo de Transacciones Electrónicas Comerciales CLUSTER\_EB

**Carlos María Chezzi**

CIDISI (UTN- FRCON)

Salta 277, Concordia, Entre Ríos

[carlos\\_chezzi@frcon.utn.edu.ar](mailto:carlos_chezzi@frcon.utn.edu.ar)

and

**Ana Rosa Tymoschuk**

**Ariel Villamonte**

CIDISI (UTN- FRSF)

Lavaise 610 Santa Fe,

[anrotym@ceride.gov.ar](mailto:anrotym@ceride.gov.ar)

[villaariel@hotmail.com](mailto:villaariel@hotmail.com)

## Abstract

The design of computer systems for sites of electronic business can be done with varied configurations of hardware and software resources and communications. Basically it has to consider the clients' requirements as regards efficient levels of attention when they make their commercial transactions. Because of that, it is convenient to have computer technological and communication platforms that assure the quality of the service with a dynamic surfing. The evaluation of the efficiency of these systems, preferably before its implementation, guides the decisions about the investments and the potential benefits. For the planning of the capacities of computer systems there are analytical methods such as the queue and queue networks theory, and their basis to obtain metrics of performance are fundamental laws and algorithms. In the case of sites of electronic business the application of these methods has certain limitations because of the complexity of these systems, mainly due to the clients' behaviour. An advisable alternative is to work with models for simulation that incorporate the representation of complexities and the dynamic analysis of the system behaviour. In this work a study case is stated and solved with the implementation of the modelling methodology for simulation of electronic transactions B2C (Business to Consumer).

**Keywords:** *Modelling. Simulation. Performance. Prediction. Electronic business. Study case.*

## Resumen

El diseño de sistemas informáticos para sitios de negocios electrónicos puede realizarse con configuraciones variadas de recursos de hardware, software y comunicaciones. Fundamentalmente debe considerar los requerimientos de los clientes en cuanto a eficientes niveles de atención al hacer sus transacciones comerciales. Por ello, es conveniente contar con plataformas tecnológicas informáticas y de comunicaciones que aseguren la calidad del servicio con una navegación dinámica. La evaluación del rendimiento de estos sistemas, preferentemente previa a su implementación, orienta las decisiones sobre las inversiones y los beneficios potenciales. Para la planificación de capacidades de sistemas informáticos existen métodos analíticos como la teoría de colas y de redes de colas, cuyos fundamentos para la obtención de métricas de performance son leyes fundamentales y algoritmos. En el caso de sitios de negocios electrónicos la aplicación de estos métodos tiene ciertas limitaciones por la complejidad de estos sistemas, fundamentalmente debida al comportamiento de clientes. Una alternativa conveniente es trabajar con modelos para simulación que incorporen la representación de complejidades y el análisis dinámico del comportamiento del sistema. En este trabajo se plantea y resuelve un caso de estudio con la implementación de la metodología de modelado para simulación de transacciones electrónicas B2C (Business to Consumer).

**Palabras Claves:** *Modelización. Simulación. Performance. Predicción. Negocios electrónicos. Caso de estudio.*

## 1 INTRODUCCIÓN

Las configuraciones de recursos de hardware y software para el diseño de sistemas informáticos han incrementado en eficiencia pero también en complejidad, como computadores con capacidades de multiprocesamiento simétrico, utilización de memorias caché, grupos de computadoras interconectados que trabajan juntas como un sistema unificado, entre otras [1],[3],[10].

Mediante la utilización de redes de alta velocidad para la interconexión de servidores y sistemas distribuidos se configuran plataformas de alto rendimiento, que aseguran la disponibilidad y funcionalidad de los negocios basados en transacciones electrónicas [3],[7].

En consecuencia, se plantean diferentes aspectos de hardware, software, comunicaciones y aplicaciones responsables de implementar las transacciones electrónicas y de este modo, no resulta simple la evaluación de las capacidades de la planificación del sistema completo.

Entre las herramientas de predicción, la teoría de colas es una técnica analítica que presenta cada recurso como una cola y propone ecuaciones para la obtención de métricas de evaluación de capacidades [6].

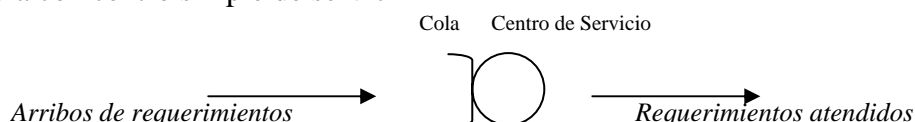
Pero las configuraciones de sistemas informáticos en general requieren de redes de recursos de diferentes características, y su tratamiento se hace mediante la teoría de redes de colas [9]. Las desventajas de estas técnicas analíticas residen en que no permiten representar en forma simple la complejidad de los sistemas de negocios electrónicos o realizar un análisis dinámico de los mismos.

La alternativa es implementar técnicas de simulación, cuya principal ventaja se debe a la capacidad de representación de sistemas reales, mediante relaciones lógicas matemáticas y la experimentación en diferentes escenarios la transforma en una herramienta de soporte de decisiones [6].

En este trabajo se propone un caso de estudio donde se implementa la metodología de modelización para simulación de transacciones electrónicas de negocios, del tipo cliente-negocio B2C [2],[7]. Se propone un modelo CLUSTER\_EB, basado en un clusters de servidores [3], agrupados en niveles de servicios (servidores web, de aplicación y de base de datos). Se plantean las cargas de trabajo sobre la base de tres clases de transacciones electrónicas, como etapa preliminar para el estudio de cargas variadas. Con el análisis de los resultados de las primeras experimentaciones se analiza el “cuello de botella” del sistema. Se proponen mejoras y se obtienen métricas e indicadores tecnológicos y métricas de negocios.

## 2 MODELIZACION

La teoría de colas se basa en el modelo general de colas formado por un centro de servicios y una cola de espera de aquellos clientes que no pueden ser atendidos. En la figura 1 se muestra un modelo de cola con centro simple de servicio.



**Figura 1. Modelo de Cola**

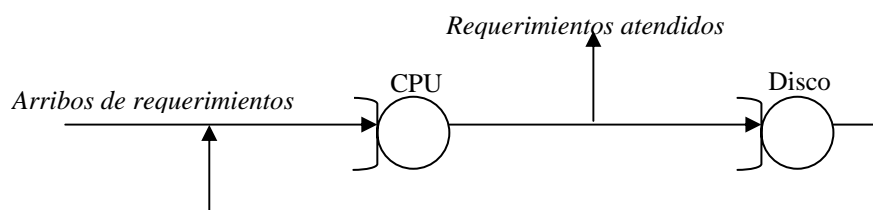
El centro de servicio tiene asignado un tiempo de servicio para atender los requerimientos que arriban según un tiempo entre llegadas. Estos pueden ser atendidos o pasados a cola de espera en caso de que el centro esté ocupado.

En los sistemas informáticos cada uno de los recursos que lo componen es análogo al funcionamiento del centro de servicio con su cola y los requerimientos de usuarios. El tiempo de respuesta de un requerimiento es la suma del tiempo que estuvo en cola y del tiempo de servicio en el centro para su atención. Según la teoría de colas así se determinan los tiempos de residencia del requerimiento en los centros de servicio [6].

El modelo de cola de la Figura 1 tiene dos parámetros: tasa de arribo de los clientes (intensidad de la carga) y requerimiento de servicio promedio de un cliente (demanda de servicio). Con ecuaciones de las leyes fundamentales se pueden obtener medidas de performance tales como utilización del recurso, tiempo promedio del requerimiento en atención (tiempo de residencia), número promedio de requerimientos en cola y velocidad de procesamiento. Es importante destacar que el número promedio de clientes en cola incluye los requerimientos en cola y en servicio [6],[9].

La teoría de colas representa al sistema completo como un solo centro. La teoría de redes de colas permite representar a cada uno de los recursos de los sistemas informáticos, conectados entre sí como una red de cola.

La Figura 2 presenta una red de colas compuesta de una CPU y un Disco. Los requerimientos arriban a la CPU y son atendidos si esta se encuentra desocupada, en otro caso quedan en cola, una vez atendidos por la CPU pasan al Disco. Atendido el requerimiento en el Disco vuelve a la CPU, para finalmente salir como requerimiento atendido.



**Figura 2. Modelo de Red de Colas**

Los parámetros de este modelo son análogos al de teoría de colas con la diferencia de que existen demandas de servicio para cada de recurso de la red. Para la obtención de las medidas de performance de los recursos y del sistema se plantean ecuaciones basadas en Leyes Fundamentales, Ley de Little, Ley de Utilización, Ley del Tiempo de Respuesta y Ley de Flujo Forzado y en algoritmos asociados a cada tipo de modelo [6].

La clasificación de modelos según:

- El sistema que representan: estáticos (no hay cambios del sistema en el tiempo) y dinámicos (se representan los cambios del sistema en el tiempo). Para sistemas dinámicos las variables podrían modificarse continuamente en el tiempo (continuos) o a pasos escalonados (discretos).
- La identificación del Número de Clientes: Abiertos (no se conoce el número máximo de clientes pero sí una tasa de arribos) y Cerrados (se conoce a priori el número total de posibles clientes).
- Carga de Trabajo: Clase Simple (un único tipo de trabajo) y Clases Múltiples (diferentes tipos de trabajo).

La modelización analítica se basa en modelos estáticos, de tiempo continuo, determinísticos o probabilísticos, con requerimientos abiertos y cerrados, para cargas de trabajos simples y múltiples.

Otra de las herramientas de modelización que permite la representación de sistemas informáticos con importantes cantidades de recursos y relaciones de funcionamiento es la simulación, mediante la cual podemos analizar el comportamiento del sistema informático desde una perspectiva dinámica.

### 3. MODELO CLUSTER-EB PARA SIMULACIÓN

El caso de estudio consiste en la simulación de un proceso de negocio electrónicos B2C, con la aplicación de la metodología basada en el formalismo y la herramienta DEVS JAVA [8]. El sistema informático tiene una configuración basada en capas: la primera capa consiste del cluster de servidores Webs, la segunda, del cluster de servidores de Aplicación y la tercera, del cluster de Base de Datos. La conexión entre capas se efectúa a través de tres redes Lans. Cada LAN se considera una conexión de tipo full duplex. El arribo de los requerimientos de Internet se hace a través del Router, quien los deriva a los clusters del Sitio de Negocios Electrónicos. Cada grupo de servidores posee un coordinador que toma los requerimientos y los asigna al servidor desocupado.

La base del diseño del modelo de simulación es la teoría de redes de colas donde el sistema informático es considerado un red de colas, cuyos servidores del cluster, redes y router son considerados colas y sus interconexiones, la estructura de la red. Cada recurso tiene asociado una demanda de servicio, que es el requerimiento de servicio promedio por clase de trabajo en el recurso. El modelo es probabilístico, de tipo abierto, y sus entradas son las velocidades de arribos de las múltiples clases de clientes, con la característica de variables aleatorias y distribuciones de probabilidad, que en este estudio se plantea como distribuciones exponenciales.

Las salidas de simulación son las métricas de performance y se basan en las leyes fundamentales y las ecuaciones de la teoría de red de colas [6].

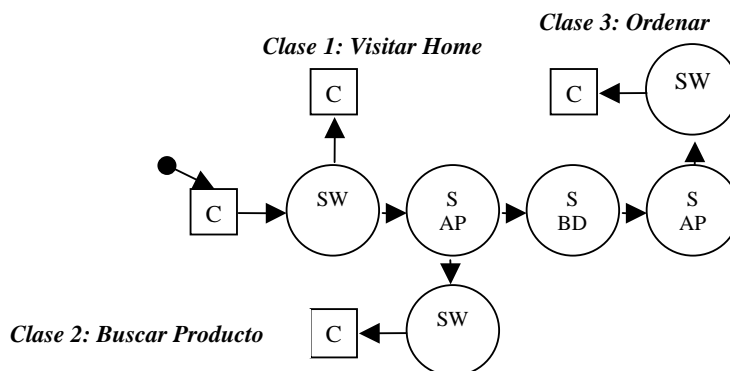
El modelo CLUSTER\_EB está representado en un diagrama UML en la Figura 3. Se organiza en dos partes fundamentales: el sistema en estudio y el marco experimental. El sistema se presenta como un acoplamiento de modelos atómicos que representan los recursos informáticos. El marco experimental se forma por los generadores de la carga de trabajo y el transductor que computa las salidas del sistema. Los acoplamientos se describen en la clase ViewableDigraph y cada modelo atómico es parte de la clase ViewableAtomic. A continuación se describen las Clases del modelo:

- *Requerimiento*: representa los requerimientos de los clientes. Contiene un identificador.
- *ClaseRequerimiento*: contiene información sobre la clase a la que pertenece un determinado requerimiento y la ruta que debe seguir el mismo. Por ruta se entiende a la lista de clusters por los cuales debe pasar el requerimiento.
- *GeneradorClase*: genera los requerimientos de una clase determinada siguiendo una distribución de tiempos entre arribos definible.
- *Transductor*: registra los requerimientos generados y completados por clase. Con esta información realiza estadísticas como tiempo promedio de respuesta, throughput, etc.
- *MEClusterRandomEB*: es el marco experimental formado por los generadores y transductores, uno por cada clase de requerimiento, el cluster web, el cluster de aplicaciones y el cluster de base de datos. Esta clase se encarga de configurar los parámetros de los clusters, los generadores, los transductores y las clases de requerimientos.
- *ClusterRandomClase*: representa un cluster. Recibe requerimientos y los procesa. Permite configurar la cantidad de procesadores y sus distribuciones de tiempos de procesamientos para cada clase de requerimiento. Está formado por un administrador de transacciones



### 3.2.- Descripción de la Carga de Trabajo.

En la Figura 4 se presenta un modelo de interacción de clientes con tres tipos de sesiones denominadas Clases 1, 2 y 3. La clase 1 visita la página principal (Home), la Clase 2 busca un producto y la Clase 3 hace la adquisición del producto. Estas interacciones dan la funcionalidad del modelo de simulación.



**Figura 4: Diagrama de Carga de Trabajo**

Donde C: Cliente, SW: Servidor WEB, SAP: Servidor de Aplicación, SBD: Servidor de Base de Datos y  $\bullet \rightarrow$ : Comienzo.

### 3.3.- Diseño de experimentos de Simulación.

La simulación se realiza ingresando como entradas los argumentos de la distribuciones exponenciales de cada clase de cliente, obteniendo así los diferentes tiempos medios entre arribos de requerimientos al CLUSTER\_EB. Se consideran los tiempos entre arribos para la clase de trabajo 1, manteniendo los tiempos de las otras dos clases constantes. Los niveles de este factor son: 0.06, 0.07, 0.08, 0.09, 0.10 segundos. Cada experiencia de simulación consiste una corrida donde se cambia la semilla de generación de variables aleatorias de 0 a 3 en la herramienta de simulación DEVS JAVA para cada nivel del factor. Cada experiencia demanda un tiempo de simulación de 500 milisegundos [5].

### 3.4.- Resultados del Modelo CLUSTER\_EB.

Para el caso de estudio se propone una plataforma tecnológica con cuatro servidores para el Cluster Web, tres servidores para el Cluster de Aplicación y dos para el Cluster de Base de Datos. Como parámetros del modelo se plantean las demandas de tiempo de servicio por cada clase de trabajo en los servidores del cluster, cuyo detalle se muestra en la tabla 1 y las correspondientes a los dispositivos de comunicaciones se muestran en la tabla 2.

**Tabla 1. Parámetros de los Servidores por Cluster para cada clase de trabajo**

Clases	Demandas de los componentes del sistema ( segundos)			
	Distribución de Tiempos entre Arribos	Cluster Web (4 servidores web)	Cluster Aplicación (3 servidores de aplicación)	Cluster Base de Datos (2 servidores de bases de datos)
1	Constante	0,712		
2	Constante	0,699	0,891	
3	Constante	0,724	0,909	1,333

**Tabla 2: Parámetros de los dispositivos de conexión y redes**

Dispositivo	Demandas de Servicio (segundos.)
Router	0.0004
LAN 1	0.01546
LAN 2	0.01546
LAN 3	0.01546

El marco experimental de los arribos de requerimientos al sistema está formado por tres generadores cuya descripción se detalla en la tabla 3, siendo variable para la clase 1. Las salidas de simulación son captadas por los transductores.

**Tabla 3. Modelo de Intensidad de carga de trabajo**

Clase de trabajo	Requerimiento	Distribución de tiempos entre arribos	T: tiempos entre arribos (segundos)
1	Visitar Home	Exponencial	0.06, 0.07, 0.08, 0.09, 0.10
2	Buscar Producto	Exponencial	3
3	Ordenar	Exponencial	15

La hipótesis de simulación es: ¿Cuál es el efecto que tiene la variación de tiempos entre arribos de la clase 1: Visitar el Home en la performance del sitio de negocios?. Para aceptar o rechazar esta hipótesis se simula variando el argumento de la distribución exponencial de la clase 1.

Sobre el escenario planteado se analiza el comportamiento del sistema según la velocidad de procesamiento y el tiempo de respuesta. Los resultados obtenidos muestran tendencias razonables, como se observa en la Tabla 4, con aumento de la velocidad de procesamiento y disminución del tiempo de respuesta según aumenta el tiempo medio entre arribos para los requerimientos de navegación (clase 1: Visitar Home):

**Tabla 4. X y R por tiempo medio entre arribos.**

Tiempo medio entre arribos: T para la clase 1 (segundos)	Velocidad de Procesamiento X (requerimientos/seg)	Tiempo de Respuesta: R (segundos)
0.06	5.47	90.88
0.07	5.43	76.18
0.08	5.39	71.33
0.09	5.36	64.94
0.10	5.34	60.77

En la Figura 5 se muestra la disminución del tiempo de respuesta por clase a medida que aumenta el tiempo medio entre arribos para los requerimientos de navegación (clase 1: Visitar Home).

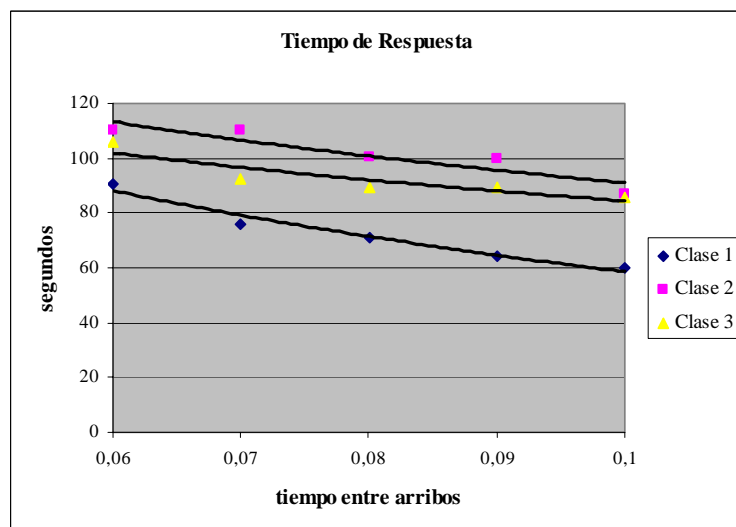


Figura 5: Tiempo de Respuesta por Clase

#### 4 ANÁLISIS DEL CUELLO DE BOTELLA.

Para encontrar el recurso que está limitando el rendimiento del sistema se realiza un análisis de detección del “cuello de botella” y las métricas en esa situación.. Este análisis se basa en las ecuaciones de límites asintóticos de la teoría de redes de colas [6], que incluye la Ley de Utilización según la ecuación:

$$U_K = \lambda * D_k;$$

donde  $\lambda$  es la tasa de arribos al sistema y  $D_k$  es la demanda de servicio del recurso k. La utilización tiende al 100 % para una determinada velocidad de arribos  $\lambda$ , llamada de saturación  $\lambda_{sat}$ , cumpliéndose que:

$$\lambda_{sat} = \frac{1}{D_{max}}$$

La última ecuación indica el límite de productividad como la frecuencia máxima de requerimientos que el sistema puede procesar. En este caso  $\lambda_{sat} = 0,75$  req./seg.

Según este análisis se calculan las utilizaciones de los recursos del sistema y se muestran en la tabla 5. Se observa que el cuello de botella está dado por los servidores del cluster web, dada principalmente por los requerimientos pertenecientes a la clase 1:

Tabla 5. Utilizaciones por cada recurso del sistema.

Recursos	Utilización (%)
Servidor Web 1	100
Servidor Web 2	99,9
Servidor Web 3	99,95
Servidor Web 4	99,91
Servidor de Aplicación 1	5
Servidor de Aplicación 2	4
Servidor de Aplicación 3	4
Servidor de Base de Datos 1	2
Servidor de Base de Datos 2	2



## 5.- MÉTRICAS E INDICADORES.

Para evaluar la performance de negocios electrónicos se deben considerar métricas e indicadores a nivel tecnológico y a nivel de negocios.

### 5.1.- Métricas a Nivel Tecnológico.

En este caso de estudio sólo se platean las métricas tecnológicas de la velocidad de procesamiento y del tiempo de respuesta.

#### 5.1.1. Velocidad de Procesamiento.

**X** = número de requerimientos de clientes completados por unidad de tiempo.

Unidad: requerimientos/segundo

#### 5.1.2. Tiempo de Respuesta.

**R** = tiempo medio de respuesta a un requerimiento de cliente.

Unidad: segundos.

### 5.2.- Indicador del Tiempo de Respuesta (IR).

Si  $R \leq 0.23$  seg. entonces IR = Sobresaliente

Si  $0.23 \text{ seg} < R < 1.77$  seg. entonces IR = Satisfactorio

Si  $R \geq 1.77$  seg entonces IR = Insatisfactorio

Para la propuesta de los valores del indicador se tomó un ranking de tiempos de respuesta de sitios de negocios reconocidos [4]. El menor tiempo de respuesta es de 0.23 seg y el promedio es de 1.77 seg.

### 5.3.- Ecuaciones de la Velocidad de Procesamiento y Tiempo de Respuesta

#### 5.3.1. Velocidad de Procesamiento:

$$X_{media} = \sum_{Clase=1}^3 X_{clase}$$

**Ecuación 2: X del sistema**

La ecuación 2 muestra que la velocidad de procesamiento del sistema es la suma las velocidades de procesamiento individuales de cada clase.

#### 5.3.2. Tiempo de Respuesta:

$$R_{media} = \sum_{Clase=1}^3 \frac{R_{clase} X_{clase}}{X_{media}}$$

**Ecuación 3: R del sistema**

El tiempo de respuesta se calculo de acuerdo a la ecuación 3, donde se observa que no es la suma de los tiempos de respuesta de cada clase o su media aritmética; sino un promedio ponderado en función de las cantidades de trabajos atendidos por clase.

### 5.4.- Métricas a Nivel de Negocios.

En términos de decisiones es conveniente pensar en un balance entre la performance de la plataforma tecnológica y el monto de la inversión. Por ello se plantea una métrica de negocios en función de la ecuación 4 del punto de equilibrio [2]:

*Costo de Inversión = Cantidad de Transacciones Ordenar x Ingreso Neto Promedio por Ventas*

#### **Ecuación 4: Punto de Equilibrio**

Tomada la métrica tecnológica X para la clase 3 (Ordenar) en un período de tiempo se obtiene la cantidad de transacciones Ordenar, que multiplicada por un ingreso promedio por ventas debe tender al Costo de Inversión para llegar al equilibrio. Un valor menor indica una pérdida y un valor mayor, las ganancias. Es importante tener en cuenta el tiempo en el cual se va a obtener este retorno.

## **6 PLANTEO DE ESCENARIOS DE SIMULACIÓN.**

De acuerdo a los resultados de la tabla 4 y del indicador del tiempo de respuesta la configuración planteada muestra una performance no satisfactoria. Para mejorar la performance se realiza un análisis en donde se tienen en cuenta dos escenarios:

Escenario 1: Mínimo Costo de Inversión: se mantienen los parámetros del sistema original y se agrega un servidor al cluster web de las mismas características que el existente.

Escenario 2: Nueva plataforma tecnológica: es una propuesta de mayor inversión, que contempla la actualización de los servidores existentes, duplicando la capacidad de cada uno de ellos y la incorporación de un servidor en cada cluster.

En los dos escenarios se mantiene la configuración de los recursos de comunicaciones.

**Tabla 6. X y R por tiempo medio entre arribos para cada escenario.**

T (clase 1) (segundos)	Escenario 1		Escenario 2	
	X (requerimientos/seg)	R (segundos)	X (requerimientos/seg)	R (segundos)
0,06	6,80	74,41	13,65	26,46
0,07	6,78	66,94	13,55	10,69
0,08	6,73	60,72	12,78	0,93
0,09	6,73	52,31	11,29	0,55
0,1	6,68	42,97	10,21	0,48

En la Tabla 6 se observa que la alternativa de menor inversión no satisface el indicador del tiempo de respuesta. En el caso del escenario 2 satisface este indicador con cargas de trabajo cuyos tiempos entre arribos T son 0.08, 0.09 y 0.1.

Sobre la base de T = 0.08 se plantea la ecuación 4 del punto de equilibrio:

$$\text{Costo de Inversión} = 0,072 \text{ req./seg.} \times \text{Ingreso Neto Promedio por Ventas } \$ \text{ seg./req.}$$

Con la estimación de los costos de inversión de la plataforma informática del escenario 2 y el ingreso promedio por ventas del negocio se analiza la tendencia al punto de equilibrio.

## **7 CONCLUSIONES**

El trabajo presenta un modelo de simulación aplicado a un caso de estudio simple, con tres clases de trabajo en un sitio web B2C, con una plataforma tecnológica conformada en cluster de servidores web, de aplicación y de base de datos, configurado en tres capas.

El modelo de simulación en DEVS JAVA permite representar la característica aleatoria y de clases múltiples de la carga de trabajo de cualquier sitio de negocios electrónicos. Con la resolución del

sistema con la herramienta de simulación DEVS JAVA se obtienen métricas que se utilizan para generar indicadores del comportamiento del sistema en cuanto al rendimiento desde el punto de vista tecnológico. Estos indicadores se relacionan con métricas de negocios para determinar los beneficios en función de las inversiones y del desempeño del sitio al momento de requerimientos de los clientes.

Los resultados presentados en este trabajo corresponden a un caso simple, previéndose en trabajos futuros la consideración de un número mayor de clases de trabajo y trayectorias aleatorias de los clientes en el sitio de negocios.

Asimismo se tiene en cuenta la necesidad de contar con otras métricas de negocios así como incorporar distintos segmentos de transacciones electrónicas, en vista a la mayor popularidad del uso de Internet para transacciones comerciales.

## REFERENCIAS

- [1] Alam, Fasihul. McNaught, Ken. Ringrose, Trevor. A comparison of experimental designs in the development of a neuronal network simulation metamodel. *Simulation Modelling Practice and Theory*. 12. Agosto 2004. 559-578.
- [2] Chezzi, C.M. Villamonte, A. Tymoschuk, A. R. Modelado y Simulación de Performance de Transacciones Electrónica Comerciales. *Proceeding IX Workshop de Ciencias de la Computación (WICC 2007)*. 443-447. Mayo, 2007.
- [3] Faour, A. Mansour, N. Weblins: A Scalable WWW Cluster-Based Server. *Advances in Engineering Software*. Vol. 37. N° 1. 11-19. Enero, 2006.
- [4][http://www.keynote.com/keynote\\_competitive\\_research/performance\\_indices/business\\_index/business.html](http://www.keynote.com/keynote_competitive_research/performance_indices/business_index/business.html)
- [5] Kleijnen, Jack P. An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research*. 164. Marzo 2004. 287-300.
- [6] Lazowska, E. Zahorjan, J. Graham, S. Sevcik, K. "Quantitative System Performance. Computer System Analysis Using Queueing Network Models". Ed. Prentice Hall Inc. (1984). Cap. 1 y 3.
- [7] Menascé, D. A., Almeida, V. A. F. *Scaling for E-Business, Technologies, Models, Performance and Capacity Planning*. Ed. Prentice Hall. New Jersey. 2000. Cap. 1, 11, 13.
- [8] Nidumolu, S. Menon, N. M. Zeigler, B. Object-oriented Business Process Modeling and Simulation: A Discrete Event System Specification Framework. *Simulation Practice and Theory*. Vol 6. N° 6. 533-571. Septiembre, 1998.
- [9] Puigjamer, R. Serrano, J. J. Rubio, Alicia. "Evaluación y Explotación de Sistemas Informáticos". Ed. Síntesis S.A. Madrid. Cap. 1, 4 y 5.
- [10] Wang, P. C. Planning experiments when some specified interactions are investigated. *Computational Statistics & Data Analysis*. 51. Febrero 2007. 4143-4151.