



UNIVERSIDAD TECNOLÓGICA NACIONAL

Facultad Regional Santa Fe

DOCTORADO EN INGENIERÍA

MENCIÓN EN INGENIERÍA EN SISTEMAS DE INFORMACIÓN

TESIS DOCTORAL

“MODELADO Y SIMULACIÓN DE DESEMPEÑO DE  
PROCESOS DE COMERCIO ELECTRÓNICO”

ING. CARLOS MARÍA CHEZZI

DIRECTORA: DRA. ANA ROSA TYMOSCHUK

CO-DIRECTOR: DR. LUCIANO PASCHOAL GASPARY

Santa Fe, Argentina.

Septiembre de 2013

Carlos María, Chezzi

Modelado y simulación de desempeño de procesos de comercio electrónico. - 1a ed. -

Concordia : el autor, 2014.

309 p. ; 0x0 cm.

ISBN 978-987-33-4311-7

1. Comercio Electrónico. I. Título

CDD 380

Fecha de catalogación: 03/01/2014

UNIVERSIDAD TECNOLÓGICA NACIONAL

Facultad Regional Santa Fe

Comisión de Posgrado

Se presenta esta Tesis en cumplimiento de los requisitos exigidos por la Universidad Tecnológica Nacional para la obtención del grado académico de Doctor en Ingeniería, mención Sistemas de Información

“MODELADO Y SIMULACIÓN DE DESEMPEÑO DE  
PROCESOS DE COMERCIO ELECTRÓNICO”

por

Ing. Carlos María Chezzi

Directora: Dra. Ana Rosa Tymoschuk

Co-Director: Dr. Luciano Paschoal Gaspar

JURADOS DE TESIS:

Dra. María de los Milagros Gutiérrez

Dra. Gabriela Beatriz Arévalo

Dr. Mario José Diván

Santa Fe, Argentina.

Septiembre de 2013



A Rosario

A mis hijos: Juanfra y Ma. Agustina



# Agradecimientos

Es difícil escribir un agradecimiento ya que son tantas las personas que me han acompañado en este desafío que seguramente olvidaré algunos nombres. Los primeros a los que les debo gratitud es a mi esposa y a mis dos hijos, para los cuales seguramente no les fue liviano seguir de cerca los momentos críticos de estudio o de trabajo. A mi directora Ana le he descubierto una gran virtud, la paciencia para perseverar junto a mí en todo momento. A ella le agradezco la formación, orientación y acompañamiento continuo. Sin mis padres y mis suegros no hubiese podido llegar a este fin, por su colaboración familiar. Cuantos ya no están y me iluminan desde el cielo, como José, Dora, los padres Ismael y Julio; mi abuela Juana y tía Santa las cuales me esperaban con un mate para los momentos de descanso. Cabe destacar la ayuda recibida de la profesora Mónica Chiovetta, como los primeros pasos de estudio en mi adolescencia. Una parte de esta camino la viví en Puerto Alegre, por ello no dejo de agradecer a Luciano, mi codirector, quien me recibió en la Facultad de Informática de la Universidad Federal de Río Grande do Sul y se comprometió también en mi formación en esa etapa. A Nina y Tania que nos ayudaron en la estadía en Brasil. Pero si hay algo que no puedo dejar de mencionar es a mi querida Facultad Regional Concordia quien me ha dado la posibilidad de realizar este desafío. Cuando menciono a la institución me refiero a Jorge, Fabián, Nidia, Agustín, Graciela, Marisa, Juan, Federico y Marcos, todas las autoridades que estimularon, sostuvieron y acompañaron. Especialmente a mis ayudantes de cátedra

Cristina y Mariela que estuvieron cuando las necesitaba. Siguiendo con nuestra UTN no dejo de agradecer a la Facultad Regional Santa Fe, a las autoridades del CIDISI en su director Dr. Omar Chiotti y a los becarios Ariel, Federico, Ricardo y Juan Manuel, quienes fueron el pilar fundamental para que pueda concretar este proyecto. A los contadores Rogelio, Luis y Mónica que asesoraron en los aspectos de negocios tratados en la tesis, así como a Oscar que me ayudó con los últimos detalles. Una dedicación especial para Pablo Gracia y Sergio Pilar por su ayuda para la definición del caso de estudio. A las profesoras de inglés Patricia, Guillermina y María Inés. A los doctores Milagros y Pablo que me respondieron siempre a las consultas que les requería. A Florencia que me dedicaba los sábados a la mañana para conversar y a Oscar que siempre tenía palabras de ayuda. A Omar que me ayudó con los detalles finales en Latex. Para terminar un especial agradecimiento a Dios que me iluminó, sostuvo y le devolvió a mi hija su salud curándola de su estado de gravedad.

# Resumen

Las transacciones de comercio electrónico requieren el soporte de tecnologías informáticas, de comunicaciones e Internet, con recursos para diseñar, entre otras, arquitecturas de servicios Web para su desempeño. Las plataformas informáticas deben asegurar la calidad del servicio con una navegación dinámica y a su vez la planificación de procesos de negocios que generen la rentabilidad necesaria para recuperar la inversión. Por esta razón es importante conocer o estimar el desempeño de los recursos tecnológicos ante la carga de trabajo generada por los clientes y los beneficios económicos comerciales mediante métricas e indicadores integrados. Las herramientas de predicción juegan un papel importante, tales como la modelización y la simulación, que en el proceso de negocios permiten comprender la esencia del sistema de negocios, identificar oportunidades para el cambio y evaluar el impacto de las variaciones propuestas sobre métricas de desempeño. La tesis plantea una metodología de modelización de las transacciones de comercio electrónico para arquitecturas orientadas a servicios, con el fin de obtener métricas integradas de negocio y tecnología mediante la simulación. En la metodología desarrollada se proponen las etapas que conducen a la construcción de modelos, comenzando con el planteo de la estrategia de negocio y la composición de transacciones del proceso de comercio electrónico. Se utiliza DEVS como framework de modelado para simulación de eventos discretos, basado en la teoría de sistemas y con recursos para modelar complejidad. El modelo representado mediante el formalismo DEVS se

implementa en la herramienta de simulación DEVSJAVA, orientada a objetos. A continuación se implementa una técnica de elaboración de los algoritmos para el formalismo DEVS para la simulación en DEVSJAVA. Finalmente se plantean diseños experimentales estadísticos para evaluar escenarios en la búsqueda de condiciones que optimicen la rentabilidad del negocio. Para demostrar la factibilidad de la metodología propuesta se implementa un modelo en base a antecedentes bibliográficos, en el cual se describen cada una de las etapas. Para la validación se desarrolla un caso de estudio de una empresa real dedicada a la venta minorista de electrónica, electrodomésticos y artículos del hogar. De este modo, se elaboran modelos de procesos de comercio electrónico cuya simulación posibilita la predicción de los beneficios económicos para una configuración de tecnología dada y con la variación de diferentes escenarios se analizan alternativas tecnológicas en relación con las posibilidades de retorno de la inversión.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Contexto . . . . .	1
1.2. Procesos de Comercio Electrónico y Medición de Desempeño . . . . .	7
1.3. Modelización, Simulación y Métodos Formales de Representación . . . . .	11
1.4. Descripción del Problema de Investigación . . . . .	15
1.5. Objetivos del Trabajo . . . . .	20
1.5.1. Objetivo General . . . . .	20
1.5.2. Objetivos Particulares . . . . .	20
1.6. Divulgaciones de los Resultados de Investigación . . . . .	21
1.7. Organización de la Tesis . . . . .	24
<b>2. Modelos de Negocios Electrónicos y de Recursos Tecnológicos y Evaluación Integrada de Desempeño</b>	<b>27</b>
2.1. Modelo de Negocios . . . . .	28
2.2. Modelo de Proceso de Negocios Electrónicos . . . . .	31
2.3. Modelo de Recursos Tecnológicos . . . . .	36
2.4. Medición de Desempeño . . . . .	37
2.5. Herramientas de Evaluación de Desempeño de Negocios y Tecnología Informática . . . . .	41

2.5.1. Aplicaciones Informáticas . . . . .	41
2.5.2. Herramientas Web para la Evaluación del Desempeño . . . . .	43
2.5.3. Planificación de la Capacidad . . . . .	44
2.6. Framework de Métricas de Desempeño de Modelos de Negocios Electrónicos	47
2.7. Simulación en Procesos de Negocios . . . . .	49
2.8. Arquitecturas Orientadas a Servicios en los Negocios Electrónicos . . .	50
2.9. Escenario de una Arquitectura Orientada a Servicios . . . . .	55
2.10. Simulación con Herramientas de Workflow . . . . .	58
2.10.1. Soporte de Workflow para Diseño de Modelos de Comercio Elec- trónico . . . . .	61
2.10.2. Modelo de Workflow del Proceso de Comercio Electrónico B2C .	63
2.11. Arquitectura de Referencia del Modelo de Negocios Electrónicos . . . .	69
2.12. Etapas de la Metodología de Modelado y Simulación . . . . .	74
2.13. Construcción del Modelo de Negocios . . . . .	76
2.14. Elaboración del Modelo de Proceso de Negocios . . . . .	77
2.14.1. Coreografía de las Interacciones con Socios . . . . .	78
2.14.2. Orquestación de Servicios Propios . . . . .	79
2.15. Planteo del Modelo de Recursos . . . . .	80
2.16. Esquematización de Métricas de Desempeño . . . . .	82
2.16.1. Enfoque Tecnológico . . . . .	83
2.16.2. Enfoque de Negocios . . . . .	85
2.17. Construcción del Modelo DEVS . . . . .	85
2.17.1. Construir el Diagrama de Estructura de las Entidades del Sistema	86
2.17.2. Delinear el Modelo Base . . . . .	86
2.17.3. Elaborar la Semántica DEVS . . . . .	89
2.18. Obtención de los Parámetros para el Modelo de Simulación . . . . .	90

2.19. Implementación en la Herramienta de Simulación . . . . .	91
2.20. Verificación y Validación del Modelo . . . . .	91
2.21. Aplicación del Algoritmo de Configuraciones Eficientes . . . . .	93
2.22. Realización del Diseño Experimental Estadístico . . . . .	97
2.23. Análisis de Resultados . . . . .	102
2.24. Conclusiones . . . . .	102
<b>3. Especificación de Sistemas de Comercio Electrónico con Técnicas Con-</b>	
<b>ducidas por Eventos Discretos</b>	<b>105</b>
3.1. Modelos de Sistemas de Eventos Discretos . . . . .	106
3.2. Diagramas de Transición de Estados . . . . .	107
3.3. Teoría de Redes de Colas . . . . .	112
3.4. Redes de Petri . . . . .	117
3.5. Statecharts . . . . .	119
3.6. Formalismo DEVS . . . . .	124
3.6.1. Formalismo DEVS Clásico . . . . .	127
3.6.2. Modelos Acoplados . . . . .	131
3.6.3. Formalismo DEVS Paralelo . . . . .	133
3.6.4. Protocolo de Simulación DEVS Paralelo . . . . .	134
3.6.5. Formalismo DEVS Estocástico . . . . .	139
3.6.6. Formalismo DEVS Paralelo y Estocástico . . . . .	142
3.7. Framework DEVS para Modelización de Transacciones Electrónicas . .	143
3.8. Marco Experimental . . . . .	147
3.8.1. Generador . . . . .	147
3.8.2. Transductor . . . . .	149
3.8.3. Modelo Acoplado del Marco Experimental . . . . .	151

3.9.	Modelo de las Transacciones de Negocios . . . . .	153
3.10.	Modelo de la Plataforma Tecnológica . . . . .	157
3.11.	Conclusiones . . . . .	163
<b>4.</b>	<b>Construcción y Simulación de Modelos de Comercio Electrónico</b>	<b>167</b>
4.1.	Descripción del Problema . . . . .	168
4.2.	Construcción del Modelo de Negocios . . . . .	168
4.3.	Elaboración del Proceso de Comercio Electrónico . . . . .	170
4.3.1.	Coreografía de Interacciones con Servicio Tercerizado . . . . .	171
4.3.2.	Orquestación de las Transacciones Internas . . . . .	172
4.4.	Planteo del Modelo de Recursos . . . . .	173
4.5.	Esquematización de Métricas de Desempeño . . . . .	174
4.6.	Construcción del Modelo DEVS . . . . .	175
4.6.1.	Diagrama de Estructura de las Entidades del Sistema . . . . .	175
4.6.2.	Modelo Base . . . . .	176
4.7.	Especificación del Modelo de Comercio Electrónico en el Formalismo DEVS	179
4.7.1.	Modelo del Marco Experimental . . . . .	179
4.7.2.	Modelo Transaccional . . . . .	180
4.7.3.	Modelo de Plataforma Tecnológica . . . . .	189
4.8.	Obtención de los Parámetros del Modelo . . . . .	193
4.9.	Implementación en DEVSJAVA . . . . .	198
4.10.	Verificación del Modelo de Simulación . . . . .	202
4.10.1.	Cálculo de Tiempo de Respuesta del Sistema . . . . .	203
4.10.2.	Cálculo de la Velocidad de Procesamiento del Sistema . . . . .	203
4.11.	Aplicación del Algoritmo de Configuraciones Eficientes . . . . .	204
4.11.1.	Realizar una Planificación de Capacidad. . . . .	204

4.11.2. Identificar Recursos Críticos . . . . .	204
4.11.3. Análisis del Indicador Económico . . . . .	209
4.12. Diseño Experimental Estadístico . . . . .	210
4.12.1. Diseño Multifactorial Completo . . . . .	212
4.12.2. Comprobación de la Idoneidad del Modelo para el Tiempo de Respuesta . . . . .	212
4.12.3. Comprobación de la Idoneidad del Modelo para la Velocidad de Procesamiento . . . . .	214
4.12.4. Análisis de Resultados de Simulación . . . . .	215
4.12.5. Métricas de Negocios . . . . .	223
4.13. Conclusiones . . . . .	225
<b>5. Caso de Estudio</b>	<b>231</b>
5.1. Descripción del Problema . . . . .	231
5.2. Modelo de Negocios . . . . .	232
5.3. Análisis del Sitio de Negocios . . . . .	233
5.4. Propuesta de Transacciones y Servicios . . . . .	236
5.5. Orquestación de Transacciones . . . . .	239
5.6. Modelo de Recursos . . . . .	240
5.7. Planteo de Métricas . . . . .	241
5.7.1. Métricas a Nivel Tecnológico . . . . .	241
5.7.2. Métricas a Nivel de Negocios . . . . .	241
5.8. Construcción del Modelo DEVS . . . . .	242
5.9. Implementación en DEVSJAVA del Modelo de Comercio Electrónico .	243
5.10. Verificación y Validación del Modelo . . . . .	246
5.11. Aplicación del Algoritmo de Obtención de Configuraciones Eficientes .	247

5.11.1. Simulación del Modelo Original para Planificación de Capacidad	247
5.11.2. Métrica de Negocios de los Escenarios 1, 2 y 3	253
5.12. Propuesta de Mejora con la Incorporación de Servicios Web	256
5.12.1. Coreografía de Servicios con Socio para Control de Inventario	257
5.12.2. Coreografía de Servicios con Socio para Cobro Electrónico	257
5.12.3. Orquestación Interna de Transacciones con Servicios Web	258
5.13. Modelo DEVSJAVA con Servicios Web	259
5.14. Simulación del Modelo con Servicios Web	260
5.15. Diseño Experimental Estadístico	266
5.15.1. Análisis de Salidas de la Variable de Respuesta Tiempo de Respuesta	267
5.15.2. Análisis de Salidas de la Variable de Respuesta Velocidad de Procesamiento	270
5.15.3. Análisis de Salidas de la Variable de Respuesta Utilización	273
5.15.4. Análisis Económico	274
5.16. Modelo de Comercio Electrónico en Capas de Servidores	275
5.16.1. Modelo DEVSJAVA de la Plataforma Informática con Servidores en Capas	278
5.16.2. Simulación del Modelo en Capas de Servidores	280
5.17. Conclusiones	283
<b>6. Conclusiones y Trabajos Futuros</b>	<b>285</b>
6.1. Conclusiones	285
6.1.1. Análisis de la metodología propuesta	285
6.1.2. Principales Contribuciones	291
6.2. Trabajos Futuros	294





# Índice de Tablas

2.1. Framework de Métricas . . . . .	48
2.2. Transacciones y recursos que las ejecutan . . . . .	81
4.1. Transacciones y responsables de las mismas . . . . .	171
4.2. Configuración de los recursos . . . . .	194
4.3. Tiempo de Servicio por Transacción para Escenario 1 . . . . .	197
4.4. Tiempo de Servicio por Transacción para Escenario 2 . . . . .	197
4.5. Tiempo de Servicio por Transacción para Escenario 3 . . . . .	197
4.6. Cargas de clientes al Sitio . . . . .	198
4.7. Tiempos de Respuesta . . . . .	203
4.8. Velocidad de Procesamiento . . . . .	203
4.9. Costos de Inversión Escenario 3 con cinco servidores . . . . .	209
4.10. Factores y niveles del experimento . . . . .	212
4.11. Variables de Respuesta . . . . .	212
4.12. Datos para el experimento . . . . .	215
4.13. ANOVA de los factores e interacciones para la variable de respuesta Tiempo de Respuesta. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados. . . . .	216

4.14. ANOVA de los factores e interacciones para la variable de respuesta Velocidad de Procesamiento. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados. . . . .	220
4.15. Desviación Estándar del Diseño Experimental Estadístico . . . . .	222
4.16. Ganancias Netas para los diferentes tiempos de servicio Web . . . . .	224
4.17. Ganancias Netas para los diferentes tiempos de servicio Web . . . . .	224
4.18. Meses para el retorno de la inversión . . . . .	224
5.1. Datos comparativos caso real y simulado . . . . .	246
5.2. Utilización del servidor del Escenario 1 . . . . .	250
5.3. Utilización del servidor para los Escenarios 2 y 3 . . . . .	252
5.4. Desviación Estándar del Tiempo de Respuesta del Escenario 1 . . . . .	252
5.5. Desviación Estándar del Tiempo de Respuesta del Escenario 2 . . . . .	252
5.6. Desviación Estándar del Tiempo de Respuesta del Escenario 3 . . . . .	252
5.7. Costos de Inversión Escenario 1 . . . . .	253
5.8. Costos de Inversión Escenario 2 . . . . .	253
5.9. Costos de Inversión Escenario 3 . . . . .	254
5.10. Productos más vendidos por día . . . . .	254
5.11. Resultado de negocios . . . . .	255
5.12. Meses para retorno de inversión en Escenarios 1, 2 y 3 . . . . .	255
5.13. Desviación Estándar del Tiempo de Respuesta del Escenario 4 . . . . .	261
5.14. Utilización del servidor del Escenario 4 . . . . .	262
5.15. Desviación Estándar del Tiempo de Respuesta del Escenario 5 . . . . .	264
5.16. Transacciones frecuentes . . . . .	265
5.17. Transacciones más frecuentes de clientes que compran . . . . .	265
5.18. Factores y niveles del experimento . . . . .	266

5.19. Variables de Respuesta . . . . .	267
5.20. ANOVA de los factores e interacciones para la variable de respuesta Tiempo de Respuesta. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados. . . . .	268
5.21. ANOVA de los factores e interacciones para la variable de respuesta Velocidad de Procesamiento. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados. . . . .	271
5.22. Costos de Inversión para tres servidores con 75 % de mayor capacidad .	274
5.23. Desviación Estándar del Diseño Experimental Estadístico. V.P. Veloci- dad de Procesamiento. T.R. Tiempo de Respuesta. . . . .	274
5.24. Tiempos de servicio por servidor . . . . .	280
5.25. Métricas tecnológicas . . . . .	281
5.26. Porcentaje de utilización por recurso . . . . .	282
5.27. Costos de Inversión . . . . .	282
5.28. Métricas de negocios . . . . .	282



# Índice de Figuras

1.1. Cantidad total de usuarios con Internet en Argentina . . . . .	2
1.2. Porcentaje de usuarios de Internet que realizan operaciones de comercio electrónico . . . . .	3
1.3. Gastos anuales por usuario de comercio electrónico . . . . .	3
1.4. Modelo de Referencia Negocios Electrónicos . . . . .	17
2.1. Capas en el modelo de negocios . . . . .	29
2.2. Tipos de Procesos Electrónicos de transacciones comerciales . . . . .	33
2.3. Arquitectura multicapa de plataforma informática de comercio electrónico	37
2.4. Proceso de planificación de capacidad . . . . .	45
2.5. Transacciones de negocios en base a servicios . . . . .	54
2.6. Arquitectura básica orientada a servicios . . . . .	55
2.7. Tecnología de servicios Web . . . . .	56
2.8. Framework de una Arquitectura Orientada a Servicios . . . . .	57
2.9. Framework de un modelo de Workflow . . . . .	59
2.10. Objetos para el diseño del modelo . . . . .	62
2.11. (a) Estructura alternativa. (b) Estructura cíclica . . . . .	62
2.12. Modelo de comercio electrónico B2C . . . . .	65
2.13. Tiempo promedio de un caso en el proceso . . . . .	66

2.14. Número de casos por tipo de sesión . . . . .	67
2.15. Tiempo promedio de un caso para diferentes frecuencias de arribos . . . . .	68
2.16. Porcentajes de utilización por servicio . . . . .	68
2.17. Vistas del Modelo de Comercio Electrónico . . . . .	70
2.18. Arquitectura de Modelización y Simulación Orientado a Servicios . . . . .	71
2.19. Etapas de la Metodología de Modelización y Simulación . . . . .	75
2.20. Coreografía del servicio de consulta . . . . .	78
2.21. Esquema general del modelo CBMG . . . . .	80
2.22. Modelo de red de colas de recursos para una consulta a una base de datos . . . . .	82
2.23. Modelo de estructura del sistema de consulta a la base de datos . . . . .	87
2.24. Modelo de estructura del sistema de consulta a la base de datos . . . . .	87
2.25. Modelo base de servicio Web . . . . .	89
2.26. Transición interna . . . . .	89
2.27. Transición externa . . . . .	90
2.28. Pasos para la generación de variables aleatorias . . . . .	91
3.1. Grafo de Transición de Estados . . . . .	108
3.2. Customer Behavior Model Graph General . . . . .	109
3.3. Ejemplo de Customer Behavior Model Graph . . . . .	110
3.4. Centro de Servicio Simple . . . . .	113
3.5. Centro de Demanda . . . . .	114
3.6. Modelo de red de colas abierta . . . . .	115
3.7. Modelo de red de colas cerrada . . . . .	115
3.8. Red de colas de un servidor Web . . . . .	116
3.9. Red de Petri de un servidor Web . . . . .	118
3.10. Statechart de un servidor . . . . .	119

3.11. Statechart del servidor Web . . . . .	121
3.12. Representación de eventos en una línea de tiempo . . . . .	126
3.13. Modelo Atómico . . . . .	130
3.14. Modelo Acoplado . . . . .	132
3.15. Proceso de Simulación DEVS . . . . .	135
3.16. Estructura AVL del Protocolo DEVS . . . . .	136
3.17. Secuencia entre coordinador y simuladores . . . . .	138
3.18. Estructura jerárquica y su modelo de simulador asociado . . . . .	139
3.19. Elementos básicos del framework DEVS . . . . .	144
3.20. Diagrama de Estructura del Proceso de Comercio Electrónico. . . . .	144
3.21. Componentes del Framework DEVS . . . . .	146
3.22. Diagrama Statechart del Generador . . . . .	148
3.23. Diagrama Statechart del Transductor . . . . .	149
3.24. Modelo Acoplado del Marco Experimental . . . . .	151
3.25. CBMG de clientes con comportamiento frecuente de compras . . . . .	153
3.26. Modelo Atómico Transaccional. . . . .	154
3.27. Modelo Atómico del Recurso Tecnológico . . . . .	158
3.28. Modelo Atómico del Recurso Tecnológico sin Cola . . . . .	160
3.29. Modelo de Red de Colas . . . . .	161
3.30. Modelo Acoplado de Plataforma Informática . . . . .	161
4.1. Diagrama de secuencia de integración con servicio de cobro . . . . .	171
4.2. Customer Behavior Model Graph de compradores frecuentes . . . . .	172
4.3. Modelo de red de colas del cluster de servidores . . . . .	174
4.4. Diagrama de estructura del modelo de comercio electrónico . . . . .	176
4.5. Modelo base de componentes . . . . .	177

4.6. Diagrama Statechart del Modelo Acoplado del Experimental Framework	180
4.7. Diagrama Statechart del Modelo Atómico de Transacción Pay . . . . .	184
4.8. Diagrama Statechart del Modelo Atómico de Web Service . . . . .	187
4.9. Diagrama Statechart del Modelo de Plataforma Tecnológica con Fallas	190
4.10. Ventana principal soapUI . . . . .	195
4.11. Ventana principal soapUI para el proyecto Paypal . . . . .	195
4.12. Ventana de la operación Direct Payment . . . . .	196
4.13. Ventana Request de la operación Direct Payment . . . . .	196
4.14. Diagrama de Clases del Modelo de Comercio Electrónico . . . . .	199
4.15. Diagrama de Clases del Modelo DEVS . . . . .	200
4.16. Ventana Simulador DEVSJAVA Modelo de Comercio Electrónico. . . .	201
4.17. Cantidad de requerimientos en cola para diferentes cantidades de servi- dores en Escenario 1. . . . .	205
4.18. Tiempo de Respuesta para diferentes cantidades de servidores en Esce- nario 1. . . . .	206
4.19. Cantidad de requerimientos en cola para diferentes cantidades de servi- dores en Escenario 2. . . . .	206
4.20. Tiempo de Respuesta para diferentes cantidades de servidores en Esce- nario 2. . . . .	207
4.21. Cantidad de requerimientos en cola para diferentes cantidades de servi- dores en Escenario 3. . . . .	208
4.22. Tiempo de Respuesta para diferentes cantidades de servidores para Es- cenario 3. . . . .	209
4.23. Residuos para el Tiempo de Respuesta en el Diseño Factorial Completo.	213
4.24. Residuos para la Velocidad de Procesamiento en el Diseño Factorial Com- pleto. . . . .	214

4.25. Tiempo de Respuesta versus factores en el Diseño Factorial Completo. .	217
4.26. Efectos de los factores para el Tiempo de Respuesta en el Diseño Factorial Completo. . . . .	218
4.27. Interacciones de los factores para el Tiempo de Respuesta en el Diseño Factorial Completo. . . . .	219
4.28. Velocidad de Procesamiento versus factores en el Diseño Factorial Com- pleto. . . . .	221
4.29. Efectos de los factores para la Velocidad de Procesamiento en el Diseño Factorial Completo. . . . .	221
4.30. Interacciones de los factores para la Velocidad de Procesamiento en el Diseño Factorial Completo. . . . .	222
4.31. Cantidad de productos vendidos por día . . . . .	223
5.1. Modelo de alto nivel del sitio de negocios de la empresa . . . . .	236
5.2. Gráfico CBMG del sitio . . . . .	237
5.3. Primera ventana en el recorrido de sesiones de clientes . . . . .	238
5.4. Ventana con la primera, segunda y tercera interacción . . . . .	239
5.5. Ventana con porcentajes de transición de estados . . . . .	239
5.6. Diagrama CBMG de la sesión de cliente en estudio . . . . .	240
5.7. Diagrama de estructura del modelo de simulación de comercio electrónico	243
5.8. Modelo de Comercio Electrónico Base . . . . .	244
5.9. Modelo DEVJAVA de la Empresa . . . . .	245
5.10. Número de clientes que arriban por día en función del tiempo entre arribos	247
5.11. Cantidad de productos vendidos por día para los diferentes tiempos entre arribos . . . . .	248
5.12. Tiempo de Respuesta Escenario 1 . . . . .	249

5.13. Velocidad de Procesamiento del Escenario 1 . . . . .	249
5.14. Tiempo de respuesta para los Escenarios 2 y 3 . . . . .	251
5.15. Velocidad de Procesamiento de los Escenarios 2 y 3 . . . . .	251
5.16. Diagrama de secuencia de integración con el proveedor . . . . .	257
5.17. Diagrama de secuencia de integración con servicio de cobro . . . . .	258
5.18. Diagrama CBMG de la sesión de cliente con servicios Web . . . . .	259
5.19. Modelo DEVSJAVA transaccional del nuevo CBMG . . . . .	260
5.20. Tiempo de Respuesta para Escenario 4 . . . . .	261
5.21. Cantidad de productos vendidos por día . . . . .	262
5.22. Tiempo de Respuesta del Escenario 5 . . . . .	263
5.23. Comprobación de Idoneidad de la variable de respuesta Tiempo de Res- puesta . . . . .	267
5.24. Gráfico de Tiempos de Respuesta . . . . .	269
5.25. Gráfico de Efectos Principales del Tiempo de Respuesta . . . . .	269
5.26. Gráfico de Interacciones del Tiempo de Respuesta . . . . .	270
5.27. Comprobación de Idoneidad de la variable de respuesta Velocidad de Procesamiento . . . . .	271
5.28. Gráfico de Velocidad de Procesamiento . . . . .	272
5.29. Gráfico de Efectos Principales de la Velocidad de Procesamiento . . . . .	272
5.30. Gráfico de Utilizaciones del Servidor . . . . .	273
5.31. Entidades del modelo de Comercio Electrónico . . . . .	275
5.32. Esquema de Servidores en Capas . . . . .	276
5.33. Diagrama de Interacción . . . . .	277
5.34. Vista del simulador DEVSJAVA del modelo de comercio electrónico con servidores en capas . . . . .	279

6.1. Etapas de la metodología de modelización para simulación . . . . . 287



# CAPÍTULO 1

## Introducción

*En este capítulo se plantea la motivación que despierta interés por el tema de investigación, se describe el problema en estudio y se elabora el marco teórico sobre la base de antecedentes bibliográficos. Una vez fundamentado el tema de tesis se proponen los objetivos a lograr con la investigación y se muestran las principales contribuciones realizadas. Por último se presenta la disposición de los contenidos de la tesis para organizar la comprensión de su lectura.*

### 1.1. Contexto

Los desarrollos de las tecnologías informáticas e Internet se han incorporado en forma activa en las actividades de las organizaciones sociales y empresariales. Indicadores de usuarios e ingresos por ventas efectuadas mediante Internet muestran el impacto económico de las transacciones de negocios efectuadas mediante este medio.

Estudios de comercio electrónico del año 2012, de la Comisión Argentina de Comercio Electrónico (CACE, 2012) revelan un 44 % de crecimiento en las ventas por esta modalidad respecto al año anterior, con montos de 15.300 millones de pesos para el

tipo B2C (Business to Consumer), 1.400 millones de pesos para C2C (Consumer to Consumer) y 180.000 millones para B2B (Business to Business).

En la Figura 1.1 se muestra el número total de usuarios de Internet entre los años 2001 y 2012, de los cuales se observa un continuo crecimiento. De 3.7 millones de usuarios existentes en el 2001 se llega a 31.1 millones en el 2012.

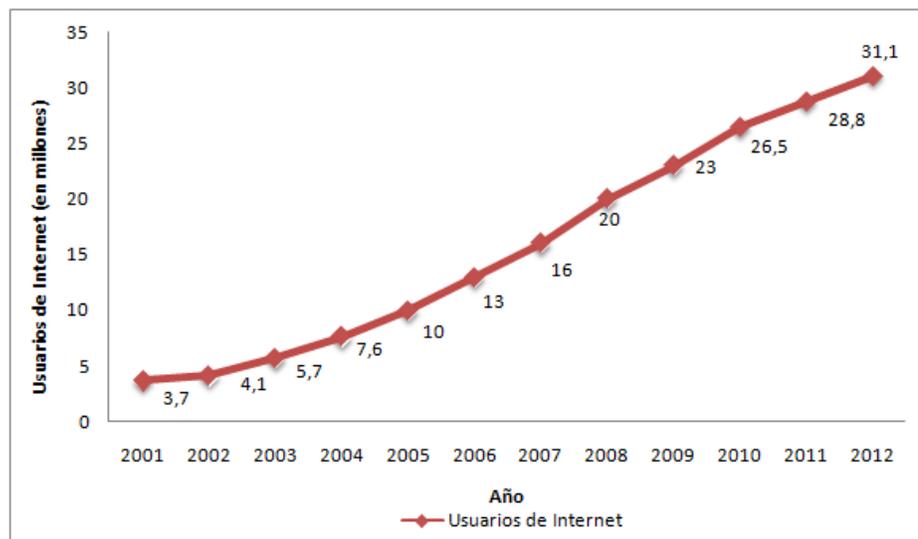


Figura 1.1: Cantidad total de usuarios con Internet en Argentina

En paralelo al crecimiento de usuarios en Internet se produce un incremento en el porcentaje de usuarios que acceden a transacciones de comercio electrónico. En la Figura 1.2 se muestra que el 11.8 % de los usuarios realizaba operaciones de comercio electrónico para el 2001, creciendo al 32.4 % en el 2012.

Los montos de compras promedios anuales por usuarios de comercio electrónico (Figura 1.3), alcanzan un monto promedio de 1.174 pesos entre los años 2004 y 2012.

La tarjeta de crédito como medio de pago sigue en crecimiento, motivado por las posibilidades de cuotas y descuento, con un incremento del 14 % respecto al año 2011. Además se observa mayor confianza por parte de los usuarios en las operaciones de

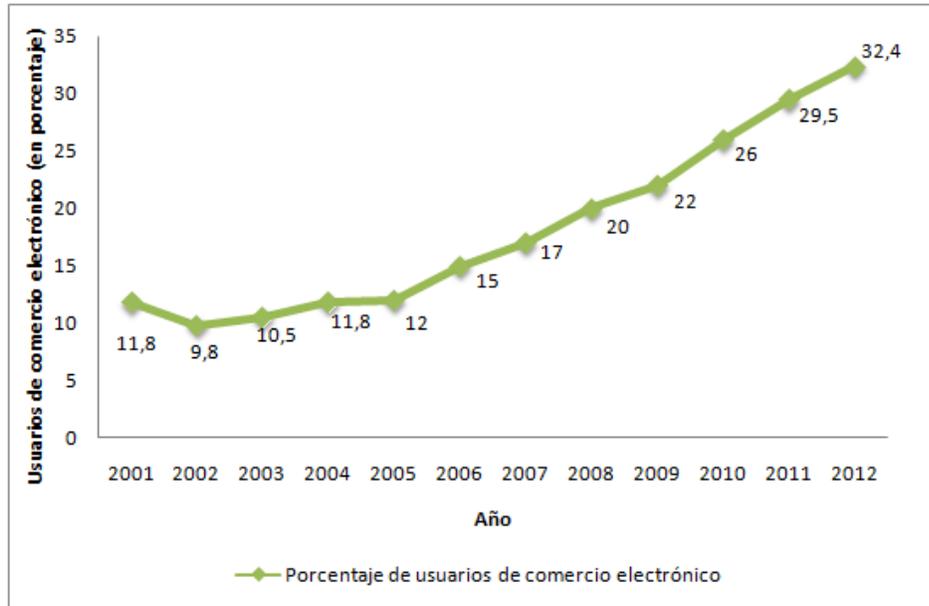


Figura 1.2: Porcentaje de usuarios de Internet que realizan operaciones de comercio electrónico

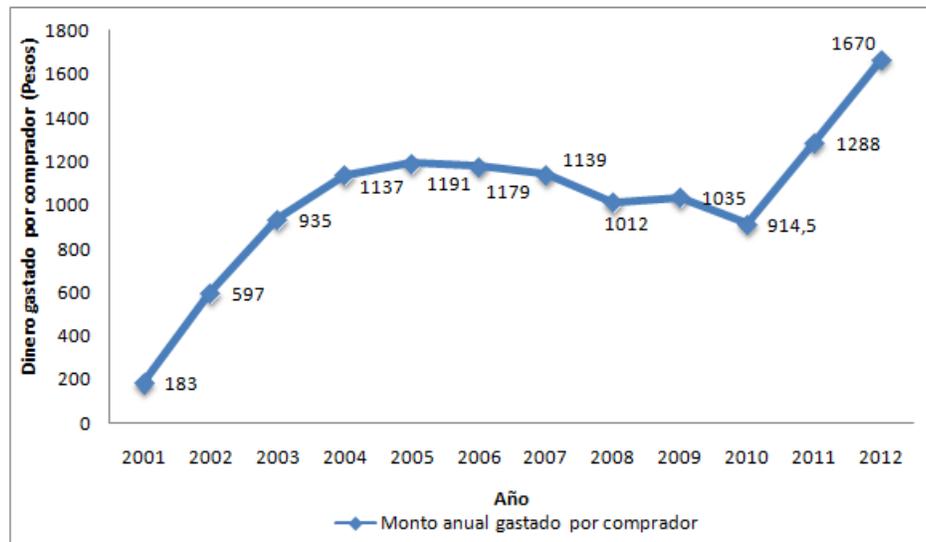


Figura 1.3: Gastos anuales por usuario de comercio electrónico

comercio electrónico.

Por tanto, los recursos ofrecidos por las tecnologías de comunicaciones e Internet se han transformado en un canal rentable de negocios, con su incorporación progresiva

en la gestión de operaciones de las organizaciones. Es así que, en la década de los años 80, la comunicación de datos se organizaba de acuerdo al estándar Electronic Data Interchange (EDI), el cual preestablecía la estructura de datos y la secuencia de mensajes a intercambiar entre computadoras. En este caso la operación de negocios se debía coordinar de modo previo a su implementación y se utilizaba para gestionar una transacción de negocios entre dos asociados.

A partir de los años 90, con el uso de Internet, la creación de la World Wide Web (WWW) y los gestores de búsqueda, se dispone de una plataforma de comunicación global donde individuos y organizaciones encuentran un escenario ideal para la ejecución de actividades comerciales (Wang y otros, 2004; Claffy y otros, 2007; Lin, 2008).

De este modo la aplicación de los aportes tecnológicos en operaciones de comercio electrónico se puede sintetizar de acuerdo a la siguiente evolución (Chu y otros, 2007):

1. *Etapas Pre-Web:* se sitúa en la década del 80 y por tanto anterior a la WWW. Se caracteriza por mecanismos de comunicación rígidos, que requieren una negociación previa del canal para llevar adelante interacciones uno a uno. Por ello, se implementan transacciones tipo Business to Business (B2B).
2. *Etapas Web Reactiva:* al principio de los años 90, a través de Internet se dispone de un ambiente de operación abierto y accesible por un mayor número de usuarios, lo cual indica el comienzo del comercio electrónico. Con la utilización de identificadores URL (Uniform Resource Locator) y navegadores de libre disponibilidad se facilita el acceso sin necesidad de un convenio de comunicación previo. Esto permite publicar información a través de los sitios Web y de catálogos, de modo que un cliente puede acceder a un portal, hacer una consulta y como reacción recibir una respuesta.
3. *Etapas Web Interactiva:* en la mitad de los años 90 se incorpora la posibilidad de

llevar adelante una negociación interactiva para la ejecución de transacciones de compra y venta. Se hace viable la implementación de comercio en línea tipo Business to Consumer (B2C) y de sitios de intermediación (Brokerage) que vinculan compradores con vendedores. Además, aparecen protocolos de seguridad como el SSL (Secure Sockets Layer) que garantizan la confidencialidad e integridad de las partes.

4. *Etapa de Integración Web:* A fines de los años 90 y comienzos del año 2000 se trabaja en la interoperabilidad de los sitios Web con sistemas internos de la empresa y entre asociados. Así surge el concepto de procesos de negocios electrónicos integrados, tales como gestión de cadenas de suministros, colaboración, reingeniería, y contratación electrónica. A través de la vinculación de las transacciones del sitio con procesos de negocios se hace posible la colaboración y el establecimiento de alianzas estratégicas.

El escenario actual se expresa en el concepto de economía digital, en la cual se requiere de los últimos desarrollos tecnológicos para responder a las presiones de la competencia, con una integración adecuada de tecnología informática y de comunicaciones a las operaciones de negocios. Como resultado de esta integración se espera reducción de costos, agregado de valor a productos y servicios, calidad en los procesos y mejores niveles de servicios en la atención al cliente (eBusiness Report, 2011).

Por tanto, la organización dispone de servicios de Internet que proponen recursos para implementar sus transacciones de comercio electrónico en línea y establecer comunicaciones para intercambiar información. Además son fuentes de creación de valor ya que permiten alcanzar un número importante de clientes, establecer comunicaciones interorganizacionales con mínimos costos y mejorar el servicio posterior a la venta con la automatización del soporte a clientes (Soto Acosta y Meroño Cerdan, 2008).

Como complemento a la automatización de transacciones, el uso de Intranets interconecta los procesos internos de la empresa y de este modo mejora las capacidades para la toma de decisiones (Lai, 2001). De este modo, la utilización de tecnologías informáticas como infraestructura de implementación de procesos de comercio electrónico muestra ser un factor de competitividad y de crecimiento de la productividad de la empresa (Giaglis y otros, 1999; Bremser y Chung, 2005).

Sin embargo la implementación de estas tecnologías no necesariamente garantiza la rentabilidad del negocio. Del análisis de la relación tecnología y desempeño de la organización y la incidencia de la información en la rentabilidad de la misma, se puede afirmar una relación positiva entre tecnologías de Internet y productividad organizacional, pero que la simple incorporación de tecnologías e información no garantiza la productividad (Soto Acosta y Meroño Cerdan, 2009).

Además, una cuestión fundamental a tener en cuenta es la estrategia de comercialización asociada a la plataforma tecnológica. Por tanto, la innovación y efectividad del uso de las tecnologías de información es resultado del plan estratégico de negocios y de una madurez organizacional en su conjunto (Cheney y Dickson, 1982).

En este contexto, se identifican tres aspectos fundamentales: infraestructura informática y de comunicaciones, modelos y estrategias propias del proceso de negocios y desembolsos de capital (Menascé y Almeida, 2000; Osterwalder y Pigneur, 2002; Bremser y Chung, 2005; Soto Acosta y Meroño Cerdan, 2008).

De acuerdo a lo expuesto, el diseño de procesos de negocios para transacciones de comercio electrónico puede realizarse con configuraciones variadas de recursos de hardware, software y comunicaciones, considerando principalmente los requerimientos de los clientes y sus expectativas en cuanto a eficientes niveles de atención y logro en las metas. Esto exige contar con plataformas tecnológicas informáticas y de comunicaciones que aseguren la calidad del servicio con una navegación dinámica (Menascé y Almeida,

1998, 2000; Menascé, 2003; Bacigalupo y otros, 2005), pero relacionadas con un costo de inversión, a recuperar en el tiempo según la rentabilidad generada.

La evaluación integrada del rendimiento de estos sistemas, preferentemente previa a su implementación, orienta las decisiones sobre las inversiones y los beneficios potenciales (Bremser y Chung, 2005; Chen y otros, 2009). La simulación en el proceso de negocios constituye una estrategia importante utilizada para comprender la esencia del sistema de negocios, identificar oportunidades para la innovación y evaluar el impacto de los cambios propuestos sobre indicadores claves de desempeño. Posibilita además experimentar decisiones de mercados con configuraciones alternativas de negocios sin interrumpir las operaciones del sistema en ejecución (Giaglis y otros, 1999).

De ahí la necesidad de contar con modelos de simulación de transacciones de comercio electrónico que midan el desempeño de sus procesos, tanto en el nivel de negocios como en el de recursos (Hahn y otros, 2002; Folan y Browne, 2005). Son diversas las situaciones y los escenarios donde la evaluación del desempeño constituye un elemento importante a la hora de las decisiones, de las inversiones y de los beneficios económicos. El conocimiento a priori del comportamiento del sistema de hardware, software y comunicaciones, y el análisis de resultados es una exigencia para garantizar la calidad de servicio y la rentabilidad del negocio.

## 1.2. Procesos de Comercio Electrónico y Medición de Desempeño

La simulación de modelos de negocios electrónicos permite estimar un conjunto de métricas clasificadas a nivel de negocios y de tecnología (Menascé y otros, 2000). La relación entre ambos grupos de métricas brinda un análisis integrado de negocios en su

modalidad electrónica.

Greasley (2000) evalúa el uso de la simulación en procesos de negocios en conjunción con herramientas de costos y presupuestos en un contexto de reingeniería. A través del modelo “as-is” y el cálculo de costos estudia el desempeño del sistema completo. Con el modelo “to-be” y la elaboración de presupuestos plantea escenarios para el análisis de sensibilidad, con el fin de reducir el riesgo de implementar soluciones ineficientes.

Hook (2011) examina la implementación de simulación en herramientas BPM (Business Process Modeling), consistente en el agregado de información adicional al modelo, tales como duración de las actividades, información de arribos y ruteo de procesos. Observa que este tipo de herramienta incorpora comportamiento dinámico como una opción extra al modelado, pero no cuenta con un algoritmo general de control de simulación. Concluye que BPM en simulación es eficiente para evaluar el desarrollo de software y mostrar el comportamiento de procesos pero aún no posee recursos para responder a cuestiones específicas de negocios.

Menascé y otros (2000) elaboran una política de simulación de servidores de comercio electrónico basada en prioridades. Para ello desarrollan un entorno experimental de simulación de un sitio Web basado en una estrategia híbrida de simulador de eventos discreto y de rastreo de eventos en tiempo real. Se simula el comportamiento de los clientes y las sesiones en el sitio Web de acuerdo a un gráfico Customer Behavior Model Graph (CBMG), se diseña un prototipo de sitio de librería electrónica que implementa las diferentes transacciones electrónicas del sitio y se originan los arribos con un generador de cargas de trabajo. La configuración de la plataforma informática que ejecuta el sitio Web se basa en una arquitectura en capas de servidores Web, aplicación y base de datos.

A partir de los arribos el simulador de sesiones de clientes funciona como un sistema adaptativo, el cual decide las transacciones de una sesión en función del estado del

cliente y el dinero acumulado en el carro de compras rastreados del sitio Web.

Respecto a las métricas, además de las tradicionales del Tiempo de Respuesta, Velocidad de Procesamiento y Utilización, se plantean dos desde una perspectiva de negocios, el monto de ingreso promedio por tiempo (dólares/segundo) y la pérdida de potenciales ingresos por clientes que seleccionan productos en su carro de compras y abandonan el sitio sin finalizar el proceso.

Debido a la complejidad de los modelos de negocios actuales, los requerimientos continuos de cambios en las estrategias y los avances de las tecnologías, Demirkan y otros (2008) replantean la arquitectura de sistemas desde una perspectiva orientada a servicios. Por tanto, las empresas deben convertir sus procesos de negocios centralizados en servicios de bajo acoplamiento y para ello elaboran un framework consistente en una estructura en niveles, cuya capa superior es la de procesos de negocios, la intermedia de servicios y la inferior de recursos de hardware y de software.

De este modo en el proceso de negocios se establece una coreografía que articula servicios que se implementan por transacciones de negocios. Los servicios y los recursos tecnológicos se corresponden con el proveedor de servicios y el proceso de negocios con el consumidor de servicios.

Oliver y otros (2008) afirman que una Arquitectura Orientada a Servicios (SOA) es adecuada para soportar los constantes cambios a los que están sujetos los procesos de negocios. Definen SOA como una arquitectura que propone una infraestructura de tecnología informática que automatiza procesos de empresas por orquestación de invocación de servicios.

Por tanto, se considera SOA como un paradigma para organizar y utilizar capacidades distribuidas que pueden estar bajo el control de diferentes dominios propietarios.

Papazoglou y van den Heuvel (2007) definen servicios como componentes computacionales (aplicaciones) autónomos e independientes de una tecnología de software,

cuya finalidad es realizar funciones que consisten desde simples operaciones hasta composición de aplicaciones distribuidas de complejos procesos.

Los servicios Web son una tecnología cuya plataforma soporta SOA. La World Wide Web Consortium (W3C) (<http://www.w3.org/>, 2012) los define como: “aplicaciones de software que pueden interactuar unas con otras, cuyas interfaces se basan en XML (eXtensible Markup Lenguaje) y están vinculadas a URIs (Uniform Resource Identifiers). La comunicación se basa en mensajes XML y protocolos de Internet”.

Kulkarni y otros (2005) presentan un caso de construcción de un sistema con SOA con la aplicación de servicios Web de Amazon. Dicha empresa se enfrenta a un crecimiento en el número de asociados, lo que la obliga a replicar la infraestructura por socio con módulos de software personalizados. A través del E-Commerce Service (ECS) abre sus aplicaciones mediante una suite de servicios Web. En esta suite los socios disponen de una infraestructura para construir y gerenciar sus sitios, implementando con invocaciones a servicios Web, búsquedas avanzadas de productos, obtención de información sobre los mismos y operación de carros de compras. De este modo Amazon reutiliza sus aplicaciones, opera canales de comunicación con socios y sostiene su liderazgo en el mercado. Por otro lado los socios encuentran una plataforma para la construcción de sus procesos de negocios, facilitando el desarrollo del sitio y minimizando los costos de inversión.

En este contexto, los procesos de comercio electrónico se enfrentan a eventos de negocios no previstos a priori y cuya aparición no siguen un comportamiento determinístico. Se requiere de un soporte de modelado para simulación orientado a servicios con capacidades de manejo de eventos asincrónicos y por ello se necesitan modelos de simulación de evento discreto (Giaglis y otros, 1999).

### 1.3. Modelización, Simulación y Métodos Formales de Representación

El término sistema es definido como la sección de la realidad, motivo de estudio, conformada por componentes integradas a través de reglas, con límites establecidos por el alcance del problema en cuestión.

Para describir y analizar el sistema se debe expresar en una forma de representación, llamada modelo. Estos modelos pueden ser clasificados como físicos, gráficos o simbólicos. La representación por modelos simbólicos se efectúa a través de parámetros y variables. Los parámetros son medidas independientes y controlables que configuran las entradas y estructura del modelo, mientras que las variables dependen de parámetros o de otras variables. Los estados de un sistema están determinados por los valores de las variables del mismo en el tiempo (Jain, 1991; Taha, 2004).

Los modelos se pueden clasificar según los cambios de estados en el tiempo como estáticos, cuando los mismos no se producen y dinámicos, con ocurrencia periódica (Cao, 1989; Khoshnevis, 1994). Si es posible predecir con certeza las salidas, el modelo es determinístico, en caso contrario el modelo es probabilístico (Jain, 1991).

El tipo de sistema considerado en este trabajo incluye los recursos de software y de hardware y los usuarios que hacen requerimientos. El sistema informático puede ser modelado por parámetros y variables que a través de relaciones representan su comportamiento, cuyas salidas no son predecibles con certeza. Entonces, se tiene en cuenta a las variables involucradas como aleatorias y con evolución en el tiempo (Menascé y Almeida, 1998, 2000). Por ello, se considera al modelo de sistema informático como simbólico, dinámico y probabilístico.

Los métodos de solución de un modelo simbólico de tipo dinámico pueden ser ana-

líticos o por experimentación en computadora. En el primer caso se requiere un razonamiento deductivo de la teoría matemática que se aplica. En sistemas informáticos se dispone de procesos de Markov, Teoría de Colas y de Redes de Cola (Lazowska y otros, 1984; Puigjaner y otros, 1992; Kleinrock, 1993; Menascé y Almeida, 1998; Agrali y otros, 2008).

En cambio el método experimental se basa en representaciones lógicas simplificadas. La simulación es una metodología experimental por computadora, cuyos modelos se fundamentan en representaciones lógicas y sus salidas se basan en experimentaciones virtuales, lo cual puede ser aplicado a modelos dinámicos (Khoshnevis, 1994).

De los métodos de simulación, se destacan (Jain, 1991):

- **Emulación:** uso de hardware o firmware.
- **Monte Carlo:** modelos probabilísticos, sin cambios en el tiempo.
- **Trace-driven:** rastreo de datos en tiempo real.
- **Evento discreto:** modelo de estado discreto.

El estudio de los sistemas informáticos consiste en la descripción del estado del sistema para una carga dada en ciertos dispositivos que lo conforman, por tanto se basa en estados discretos y para trabajar con ellos es adecuada la simulación de evento discreto (Khoshnevis, 1994).

El modelado de sistemas informáticos para resolución analítica con procesos de Markov y Teoría de Colas estudian al sistema completo sin identificar cada recurso físico en particular (Zhang y Fan, 2008). La teoría de redes de colas (Squillante y otros, 2002) permite representar a cada uno de los recursos de los sistemas, conectados entre sí como una red de colas (Lazowska y otros, 1984; Puigjaner y otros, 1992; Baldwin y otros, 2003). Si bien los métodos analíticos tienen la ventaja de exactitud en los resultados,

sus modelos presentan limitaciones porque no se pueden aplicar cuando no se cumplen ciertas hipótesis en situaciones tales como posesión simultánea de recursos por parte de algunos procesos y prioridades, entre otras. A su vez, no posibilitan la representación de sistemas complejos. La alternativa para resolver estos casos es utilizar modelos de simulación de evento discreto (Giaglis y otros, 1999; Hollocks, 2001; Alam y otros, 2004; zu Eissen y Stein, 2006; Feng y Zhang, 2008).

Para representar modelos para simulación se dispone de métodos formales. Estos métodos permiten describir el comportamiento de procesos con base en una definición formal de la estructura sintáctica y una semántica fundamentada en el álgebra. De la aplicación del método formal a la solución de un problema se obtiene un algoritmo bien conformado, base para la escritura de la aplicación en un lenguaje. Entre ellos, existen algunos como Cálculo  $\pi$  (Puhmann y col, 2005), Redes de Petri (Balbo y otros, 2000) y DEVS (Discret Event System Specification) (Zeigler y Sarjoughian, 2003; Wainer, 2003).

Cálculo  $\pi$  y Redes de Petri son dos métodos que demuestran fortalezas para la formalización de la lógica de procesos. Cálculo  $\pi$  es un técnica de representación de procesos y se basa en el concepto de movilidad que incluye comunicación y cambio, de modo que la estructura de los procesos cambian en el tiempo por comunicación de mensajes. Mientras que las Redes de Petri son una técnica de representación gráfica de sistemas de evento discreto basada en tres elementos básicos: estados, transiciones y relaciones de flujo.

Tanto Cálculo  $\pi$  como Redes de Petri son un álgebra por tanto se definen por una estructura y un conjunto de reglas asociadas a la misma. Destacando para el primer método, robustez para la construcción de razonamientos con capacidades de representación de comportamiento dinámico a través de interacciones basadas en el intercambio de mensajes. Para el segundo método, la disponibilidad de una colección de patrones de

workflow, que demuestran la factibilidad de su utilización en modelización de procesos (van der Aalst W. M. P., 2006).

Por su propia complejidad, la modelación de sistemas informáticos implica un gran número de estados e interacciones que no puede ser representada por los dos métodos anteriores.

Bernard Zeigler, del Arizona Center for Integrative Modeling and Simulation (ACIM) de la Universidad de Arizona, introduce en 1972 el framework DEVS para el modelado y la simulación de eventos discretos, donde se considera que el sistema cambia de estado por ocurrencia de eventos aleatorios y se basa en los principios de ingeniería en sistemas, constituyendo además un álgebra.

DEVS es un formalismo con capacidades de diseño de modelos en el paradigma orientado a objetos y es empleado en la construcción de modelos de simulación de sistemas dinámicos con eventos discretos. Su metodología centrada en el modelado jerárquico y modular permite representar complejidad con jerarquía de componentes, escalabilidad y reusabilidad (Zeigler y Sarjoughian, 2003).

A través de modelos atómicos se representa cada componente del sistema informático, el que se puede interconectar con otros modelos atómicos, en forma modular y jerárquica. Un conjunto de modelos atómicos se constituye en un modelo acoplado o sistema más complejo, que a su vez termina cumpliendo la función de un nuevo modelo atómico que se interconectará con otros y así sucesivamente (Zeigler, 1976). Una ventaja importante es que el método de simulación es independiente del modelo, de modo que se construye un modelo de sistema en cuestión sin estar sujeto a las restricciones de un simulador. Hace posible el modelado de aplicaciones, sistemas distribuidos y de producción (de Gentili y otros, 2005).

Castro y otros (2010) proponen el STDEVS (stochastic DEVS) en el cual incorporan funcionalidades para simulación estocástica y demuestran sus bondades para el

modelado de sistemas con comportamiento no determinístico.

Sarjoughian y otros (2008) elaboran un framework de simulación SOA-compliant DEVS (SOAD) para modelado de sistemas orientados a servicios, el cual se compone de modelos atómicos que representan al proveedor y consumidor de servicios y sus interacciones a través del paso de mensajes entre ellos. De la composición de estos servicios se generan los modelos de software orientados a servicios.

DEVJSJAVA (Nidumolu y otros, 1998; Zeigler, 2003; Sarjoughian y otros, 2000; Nidumolu y otros, 1998) es una herramienta de software orientada a objetos, cuyo lenguaje de programación es JAVA, es de libre disponibilidad para aplicaciones académicas con acuerdo de licencias, permite implementar los modelos DEVS y dispone de un marco experimental y salidas gráficas (Palaniappan y otros, 2006). Como extensión de DEVJSJAVA, Kim y otros (2009) desarrollan DEVS-Suite como un entorno de visualización de datos y con funcionalidades para el diseño experimental de la simulación.

De este modo es necesario plantear nuevos modelos de negocios electrónicos en una SOA y elaborar una metodología de simulación, con capacidades predictivas, implementables en una herramienta de simulación que obtenga métricas de negocios integradas a las capacidades tecnológicas como herramienta para la toma de decisiones comerciales.

## 1.4. Descripción del Problema de Investigación

Los problemas de desempeño de procesos de negocios en Internet presentan características impredecibles debido al comportamiento no controlable del volumen de tráfico. Además, deben responder con calidad de servicio a los requerimientos de los usuarios.

Por ello, la mejora de desempeño del negocio electrónico está asociada a la inversión en tecnología y por tanto con sus costos, beneficios y riesgos (Love y otros, 2004). Previo a la implementación de los negocios electrónicos o a sus mejoras tecnológicas es con-

veniente evaluar los costos asociados con aplicaciones, recursos de hardware y cambios organizacionales, en comparación con los beneficios como retorno de la inversión.

Hahn y otros (2002) consideran que la evolución del comercio electrónico ha superado la idea del espacio virtual para atraer clientes, distinguiendo una etapa de madurez fundamentada en la habilidad para conducir operaciones en línea, con el análisis de rentabilidad asociada.

Para posibilitar una adecuada implementación de procesos de negocios basados en transacciones electrónicas es necesario contar con herramientas tecnológicas que posibiliten la evaluación del desempeño de los mismos con el fin de predecir comportamientos y evaluar retornos de las inversiones.

El Modelo de Referencias de Negocios Electrónicos (Menascé y Almeida, 2000) plantea cuatro capas subdivididas en dos grupos principales. En la Figura 1.4 se observa el grupo de alto nivel con un enfoque de negocios y el grupo de bajo nivel con una orientación al enfoque tecnológico.

Los dos bloques superiores se concentran en el proceso de negocios y en las transacciones del sitio que responden a las metas según la estrategia organizacional. Con respecto a los bloques inferiores, se proponen el modelo de carga de clientes y de infraestructura tecnológica que soporta la implementación del negocio electrónico.

De este modo se contemplan las transacciones que llevan adelante las estrategias del modelo de negocios, el comportamiento de clientes en su navegación por el sitio y la infraestructura tecnológica que describe y caracteriza las principales componentes de hardware y de software. El desempeño de este modelo de referencia puede evaluarse por métricas de negocios y tecnológicas en forma integrada.

Además, la organización en capas facilita el planteo de requerimientos en el diseño tales como: (i) disponibilidad de servicio, que asegure la prestación sin interrupciones; (ii) escalabilidad, que permita expandirse frente a nuevos requerimientos sin reempla-

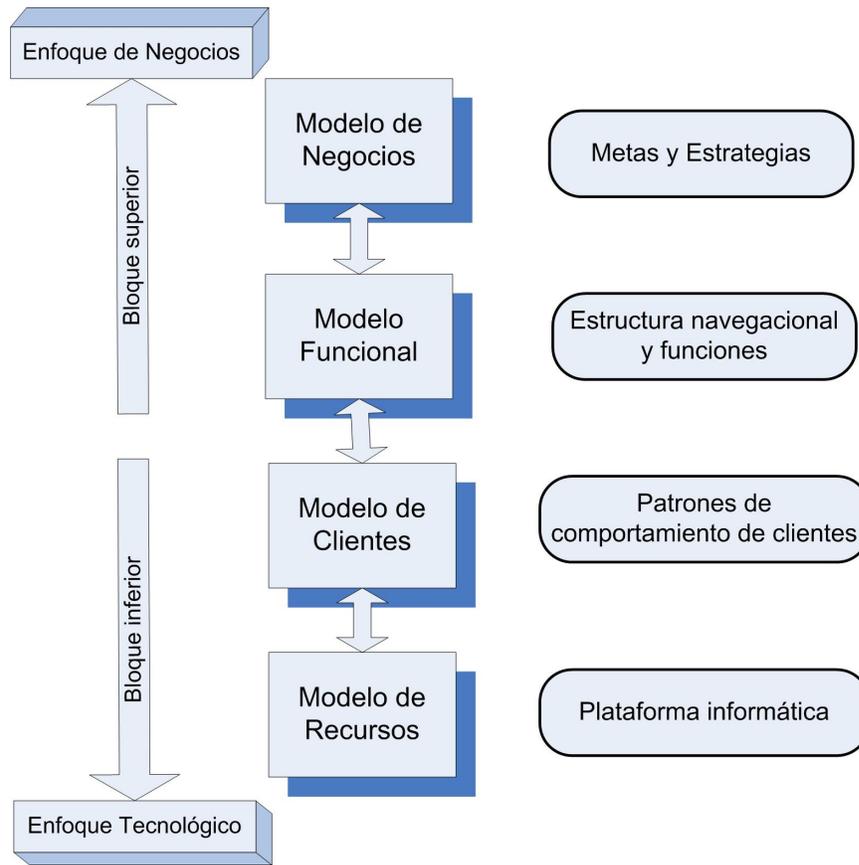


Figura 1.4: Modelo de Referencia Negocios Electrónicos

zar el sistema existente en su totalidad; (iii) rendimiento, alcanzando capacidades de cómputo que ejecuten cargas pesadas con eficiencia y (iv) flexibilidad frente a cambios.

En este sentido es necesario modelar las estrategias del negocio para obtener métricas del rendimiento económico en función de la plataforma tecnológica y las funcionalidades del sitio. A su vez se deben analizar los indicadores para cada métrica, que orienten hacia acciones de mejora de desempeño a nivel de recursos y de negocio.

Las métricas vinculadas al bloque de tecnología evalúan la infraestructura de recursos informáticos y de comunicaciones y las relacionadas con el proceso de negocios miden los beneficios económicos, que por la naturaleza de las transacciones vía Internet, tienen relación directa con la infraestructura tecnológica.

De ahí la importancia de conocer el desempeño de los procesos de comercio electrónicos a nivel de negocios en función de los recursos tecnológicos que lo implementan. Es un desafío la evaluación integrada entre la interacción de las capas correspondientes al enfoque tecnológico como soporte de las funcionalidades requeridas para la gestión del proceso de negocios.

La planificación de métricas requiere del diseño de un framework de formalización. Osterwalder y Pigneur (2002) destacan cuatro áreas fundamentales para el diseño de un framework de métricas de modelos de negocios electrónicos: definición del producto y proposición de valor al mercado, interface con el cliente; infraestructura, logística y redes empresariales para la gestión y aspectos financieros (modelo de ingresos y estructura de costos).

El aspecto financiero es considerado transversal al resto de las áreas y su objetivo es identificar la rentabilidad del negocio. El modelo de ingresos mide la habilidad de la empresa para lograr beneficios y el de costos computa los gastos de la creación de valor a los productos o servicios.

Otra característica a considerar es el nivel de vinculación presentado por modelos de organizaciones con el fin de lograr los objetivos de negocios, que se manifiesta en redes de socios, infraestructura de tecnología informática distribuida, interconexión de empleados, comunicación en línea con el cliente, entre otras (Abuosba y El-Sheikh, 2008).

Plataformas Navegador y Servidor son tecnologías de implementación comúnmente utilizadas para soportar aplicaciones en la Web. Sin embargo, en negocios electrónicos se observa una baja utilización de recursos, indicada entre el 5 % y 20 % de la capacidad total. Esto motiva a optar por un cambio en el paradigma de diseño de procesos. Se deja de percibir las aplicaciones y equipamiento como entidades tecnológicas para concebirlas como servicios de negocios (Castro Leon y Jackson, 2009). Además, las aplicaciones

tradicionales están sujetas a una herramienta de programación específica, centradas en servidores que dificultan invocación distribuida (Mulik y otros, 2008).

Se necesita unificar soluciones en una red global y orquestar la ejecución de operaciones en un ambiente transparente, con una arquitectura dinámica y eficiente que integre plataformas heterogéneas con independencia del sistema operativo y lenguaje de programación. Por ello, se debe pensar en una metodología basada en servicios, que permita vincular aplicaciones (Kulkarni y otros, 2005).

SOA es un estilo arquitectural y su fundamento metodológico es la interacción entre proveedor y consumidor de servicios con un acoplamiento débil e interfaces estándares, cuya finalidad es lograr procesos de negocios ágiles (Papazoglou y van den Heuvel, 2007). De la definición se destaca:

- **Estilo arquitectural:** medio de planificación, diseño, implementación y gestión de sistemas informáticos.
- **Pobre acoplamiento:** servicios independientes de su sistema operativo o lenguaje de programación que lo implementa, transparencia respecto a la ubicación, interacción a través de mensajes asincrónicos y descripción estandarizada de interfaces y su semántica.
- **Agilidad en procesos de negocios:** a través de composición de aplicaciones internas o externas a la organización.

El beneficio de SOA es la concepción de aplicaciones como servicios, su reusabilidad e independencia de una tecnología específica y la capacidad de ejecutar funciones de negocios. Los servicios Web son una tecnología cuya plataforma soporta SOA.

En resumen, las aplicaciones de negocios basados en transacciones electrónicas pueden ser modeladas con SOA e implementadas con tecnologías servicios Web, lo que

posibilita integración de operaciones de negocios en Internet a través de interfaces y estructuras de datos estándares.

De este modo es necesario plantear nuevos modelos de negocios electrónicos con una arquitectura orientada a servicios y elaborar una metodología de simulación, con capacidades predictivas, implementables en una herramienta de simulación que obtenga métricas de negocios integradas a las capacidades tecnológicas como herramienta para los negocios.

## 1.5. Objetivos del Trabajo

### 1.5.1. Objetivo General

El objetivo general de la tesis es desarrollar una metodología de modelado y simulación de sistemas de eventos discretos para transacciones de comercio electrónico, con el fin de evaluar el desempeño a nivel de negocio y de recursos tecnológicos y predecir comportamientos o cambios en la configuración del sistema, para obtener la rentabilidad de las inversiones.

### 1.5.2. Objetivos Particulares

Para alcanzar el objetivo general planteado, se proponen los objetivos específicos que se detallan a continuación:

- Formular modelos conceptuales de comercio electrónico (de alto nivel) y de sistemas informáticos y comunicaciones (de bajo nivel).
- Diseñar un Framework de Métricas de Desempeño que contemple la integración de los requerimientos tecnológicos y de negocios, como herramienta de evaluación

de una inversión.

- Desarrollar una metodología de modelado y simulación de procesos de comercio electrónico con el fin de obtener métricas de desempeño tecnológicas y de negocios.
- Definir un mecanismo de verificación y validación de los modelos de simulación con base en la experimentación.
- Elaborar un caso de estudio que implemente la metodología propuesta.

## 1.6. Divulgaciones de los Resultados de Investigación

Los resultados del trabajo realizado en esta tesis han sido divulgados a través de las siguientes publicaciones, exposiciones y aplicaciones en trabajos de investigación.

### **Publicaciones con Referato**

- Chezzi, C. Ma. Tymoschuk, A. R. Lerman, R. A Method for DEVS Simulation of E-Commerce Processes for Integrated Business and Technology Evaluation. Society for Computer Simulation International San Diego, CA, USA 2013. Art. N° 13, ACM. ISBN: 978-1-62748-032-1.

### **Presentaciones y Publicaciones con Referato en Proceeding de Congresos**

- Chezzi, C. Ma. Tymoschuk, A. R. Lerman, R. A Method for DEVS Simulation of E-Commerce Processes for Integrated Business and Technology Evaluation. 2013 Spring Simulation Multi-Conference. Symposium on Theory of Modeling and Simulation DEVS Integrative M&S Symposium (TMS/DEVS 2013). 7 al 10 de abril de 2013. San Diego, California, EEUU.

- Cámara, J. M. Lerman, R. Baroni, F. Chezzi, C. Ma. Tymoschuk, A. R. Modelo de Simulación de Sistema de Comercio Electrónico para Evaluación de Desempeño. 42 Jornadas de Informática e Investigación Operativa (JAIIO 2013). Simposio de Investigación Operativa (SIO 2013). 12 al 20 de septiembre de 2013. Córdoba.
- Chezzi, C.M. Lerman, R. Baroni, F. Tymoschuk, A. R. Modelo Devs para simulación integrada de transacciones electrónicas y plataforma informática. Proceeding XXXVII Conferencia Latinoamericana de Informática (CLEI 2011). 10 al 14 de octubre de 2011. Quito (Ecuador).
- Baroni, F. Chezzi, C.M. Tymoschuk, A. R. Herramienta BPEL para el desarrollo de Aplicaciones de Comercio Electrónico con Servicios Web. Proceeding XII Workshop de Ciencias de la Computación (WICC 2010). 5 y 6 de mayo, 2010. EL Calafate, Santa Cruz
- Chezzi, C.M. Tymoschuk, A. R. Workflow Support of Service Oriented B-to-C Transaction. Proceeding XV Congreso Argentino de Ciencias de la Computación (CACIC 2009). Jujuy. Octubre de 2007. ISBN: 978-897-24068-4-1.
- Chezzi, C.M. Tymoschuk, A. R. Villamonte, A. Herramienta para Análisis de Configuraciones de Sistemas Eficientes de e-Business. Proceeding XXXIV Conferencia Latinoamericana de Informática (CLEI 2008). Proceeding XXXVII Jornadas Argentinas de Informática e Investigación Operativa (JAIIO 2008). 4 al 8 de septiembre de 2008. Santa Fe (Santa Fe). 949-958.
- Chezzi, C.M. Villamonte, A. Tymoschuk, A. R. Modelo de Simulación de Transacciones Electrónicas Comerciales. Proceeding I Encuentro Regional Argentino Brasileño de I.O. XXI ENDIO 2008 Encuentro Nacional de Docentes en Investigación

Operativa. XIX EPIO Escuela de Perfeccionamiento in Investigación Operativa. 21 al 23 de mayo de 2008. Posadas (Misiones).

- Chezzi, C.M. Villamonte, A. Tymoschuk, A. R. Simulación del Modelo de Transacciones Electrónicas Comerciales CLUSTEREB. Proceeding XIII Congreso Argentino de Ciencias de la Computación (CACIC 2007). Proceeding XII Congreso Argentino de Ciencias de la Computación (CACIC 2007). 1 al 5 de octubre de 2007. Chaco y Corrientes. ISBN: 978-950-656-109-3, pp. 507-517
- 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). 3 y 4 de mayo, 2007. Trelew, Chubut. ISBN: 978-950-763-075-0, 443-447.

### **Exposiciones en Jornadas**

- Chezzi, C. Ma. Tymoschuk, A. R. Lerman, R. A Method for DEVS Simulation of E-Commerce Processes for Integrated Business and Technology Evaluation. 42 Jornadas de Informática e Investigación Operativa (JAIIO 2013). Simposio de Investigación Operativa (SIO 2013). 12 al 20 de septiembre de 2013. Córdoba.
- Chezzi, Baroni, Federico, A. Tymoschuk, A. R., C.M. Villamonte. Study and Analysis of B2C Transactions Performance with DEVS Simulation. XV CLAIO - ALIO-INFORMS Joint International Meeting. Organizado por: Association of Latin-Iberoamerican Operational Research Societies, Institute for Operations Research and the Management Sciences. 6 al 9 de junio de 2010. UBA. Buenos Aires. Exposición.

## Trabajos en Jornadas de Jóvenes Investigadores generados desde el tema de tesis

- Ricardo Lerman. Modelo DEVS de plataforma informática para simulación de transacciones de comercio electrónico con incidencia de fallas en servidores. XIV Encuentro de Jóvenes Investigadores de la Universidad Nacional del Litoral (UNL) y V Encuentro de Jóvenes Investigadores de Universidades de Santa Fe. 3, 4 y 5 de noviembre de 2011. Facultad Regional Santa Fe. Ingeniería en Sistemas de Información.
- Federico Baroni. Ricardo Lerman. Desarrollo de aplicaciones de comercio electrónico utilizando servicios Web y herramientas BPEL. Congreso Nacional de Estudiantes de Ingeniería en Sistemas de Información CNEISI 2010. Facultad Regional Santa Fe. Santa Fe, 2010.
- Federico Baroni. Plataforma tecnológica para el diseño de transacciones de negocios electrónicos B-to-C con servicios Web. XIII Encuentro de Jóvenes Investigadores de la Universidad Nacional del Litoral (UNL) y IV Encuentro de Jóvenes Investigadores de Universidades de Santa Fe. 19 y 20 de noviembre de 2009. Facultad Regional Santa Fe. Ingeniería en Sistemas de Información.

## 1.7. Organización de la Tesis

En el Capítulo 1 se presentó la Introducción a la tesis, en el cual se describió el tema de investigación y los antecedentes bibliográficos que fundamentan el problema en estudio. Para elaborar los modelos de simulación de eventos discretos se propone una metodología a partir de la cual diseñar los modelos de comercio electrónico y simular

su comportamiento con el fin de evaluar el desempeño desde una perspectiva integrada de negocios y tecnología, por lo cual la tesis se organiza en los siguientes capítulos.

En el Capítulo 2 se plantean los conceptos de modelo de negocios, proceso de negocios que implementa el modelo de negocios y procesos de comercio electrónico. Se introduce en la medición de desempeño de modelos de comercio electrónico y se describe una arquitectura orientada a servicios como referencia para el diseño.

Se plantean diferentes técnicas y herramientas de evaluación de desempeño a nivel de negocios y tecnología, se propone la simulación como estrategia de análisis del desempeño y se muestra un caso de aplicación en una herramienta de Workflow.

Como resultado de este capítulo se elabora un framework de métricas de desempeño y se propone la metodología de modelado para simulación de transacciones de comercio electrónico orientada a servicios. El framework es un instrumento de referencia para la evaluación de la productividad del negocio en función de la tecnología que lo implementa y análisis de las posibilidades del retorno de la inversión.

En cuanto a la metodología se describen las diferentes etapas, de las cuales se destacan: definición del modelo de negocios para una estrategia con transacciones electrónicas, elaboración del proceso de negocios a través de la composición de servicios o interacción de transacciones, selección de métricas integradas de negocios y tecnología, construcción del modelo DEVS, evaluación de configuraciones eficientes por simulación, diseño experimental estadístico y análisis de salidas.

En el Capítulo 3 se presentan las teorías y herramientas para modelización de sistemas de eventos discretos, las cuales se muestran en el diseño de sistemas informáticos de procesos de comercio electrónico para la evaluación de desempeño. Se trabaja con diagramas de transición de estados, redes de colas, redes de Petri, diagramas statechart y el formalismo DEVS. Respecto al último se introduce el clásico y se consideran sus variantes para el caso paralelo y estocástico.

Con base en DEVS estocástico y paralelo se definen las principales entidades del modelo de comercio electrónico y se diseñan diagramas statechart de cada componente. Siguiendo la técnica de construcción de diagramas statechart se induce el formalismo DEVS, con la identificación de puertos de entrada y de salida, estados, funciones de transición de estados, función de avance del tiempo y acoplamientos entre componentes. Se obtiene así una representación formal de la cual se confecciona la semántica del modelo de simulación.

En el Capítulo 4 se aplica la metodología en un modelo de comercio electrónico tipo B2C sobre la base de datos bibliográficos, con el fin de desarrollar cada etapa y mostrar los detalles que orientan la implementación. Se construye el modelo DEVS de un proceso de comercio electrónico para la venta minorista de libros, se lo implementa en DEVSJAVA y se muestra el modo de obtención de configuraciones eficientes de negocios en base al estudio de escenarios de tecnología informática y su relación con las posibilidades del retorno de la inversión.

En el Capítulo 5 se elabora un caso de estudio de una empresa dedicada a la venta minorista de electrodomésticos, electrónica, computación, artículos para el hogar, música y videos. Su construcción se basa en datos reales obtenidos de la monitorización del sitio de comercio electrónico de la empresa. Se elabora un gráfico CBMG de clientes, se obtienen los parámetros y se validan los resultados de simulación en contraste con los del sistema real.

Finalmente, en el Capítulo 6 se presentan las conclusiones y se analizan trabajos futuros.

## CAPÍTULO 2

# Modelos de Negocios Electrónicos y de Recursos Tecnológicos y Evaluación Integrada de Desempeño

*En este capítulo se presentan los conceptos de modelo de negocios, proceso de negocios y su especificación para tipos de modelos con estrategias electrónicas. A partir de estos conceptos se analizan técnicas de evaluación de desempeño desde una visión de negocios y tecnológica. Se propone un framework de métricas para la evaluación integrada de desempeño tecnológico y de negocios, con el fin de orientar la selección de las variables de respuesta. De este modo a partir de la inversión en tecnología se mide su productividad y evalúa posibilidades de retorno de la misma. Con base en lo antes desarrollado se presenta y describe la metodología de modelización y simulación de procesos de comercio electrónico. La construcción del modelo de simulación se sustenta en una arquitectura orientada a servicios y su implementación en el formalismo DEVS. Se elabora un framework de referencia para modelado y simulación, se proponen las etapas y se detalla un algoritmo para la obtención de configuraciones eficientes.*

## 2.1. Modelo de Negocios

El escenario de operaciones de negocios se desenvuelve en un ambiente competitivo, con tecnologías de información y comunicaciones en constante avance e incorporación en todos los estratos de la empresa. Además el ciclo de vida de productos o servicios es cada vez más corto y la dimensión del mercado es mayor debido a la globalización.

Esto obliga a gerentes a enfrentarse con situaciones de decisiones para sostener una continua proposición de valor a sus productos o servicios, selección adecuada de asociados y configuración de redes que disminuyan los costos de operaciones e incrementen el número de clientes. La complejidad que provoca la cantidad de variables en cuestión se puede abordar mediante la construcción de modelos de negocios.

Para comprender el concepto de modelo de negocios, se efectúa una búsqueda bibliográfica, de la cual se destacan las siguientes definiciones:

- Huang y otros (2011) lo definen como métodos y supuestos que hacen explícita la creación de valor en un negocio y el modo de lograr beneficios económicos en un ambiente competitivo. Una arquitectura lógica que describe productos y servicios, así como los actores involucrados y sus roles.
- Para Doganova y Eyquem-Renaultb (2009) es una descripción de la lógica de creación de valor de una empresa y su estructura, con el fin de explotar oportunidades de negocios.
- Osterwalder y otros (2005) expresan: “Es una herramienta conceptual que especifica componentes de una organización y sus relaciones para trazar su arquitectura, redes de socios y lógica de ingreso de dinero, exponer el valor que se ofrece a clientes y de este modo demostrar flujos de ingresos sustentables”.

De acuerdo a los conceptos antes expuestos se realiza la siguiente definición:

*El modelo de negocios es una herramienta conceptual que precisa la estructura de una organización, formula sus metas, describe el producto o servicio a ofrecer, con el fin de representar una arquitectura lógica que especifica las principales transacciones, información requerida, redes de asociados e identifica actores y asigna roles, analizando oportunidades de negocios y planificando fuentes de ingresos de dinero.*

Como complemento a la definición se plantea el modelo de negocios en la Figura 2.1 con una representación en tres capas, una superior llamada Estratégica, otra intermedia del Modelo de Negocios y la inferior dirigida al Proceso de Negocios.

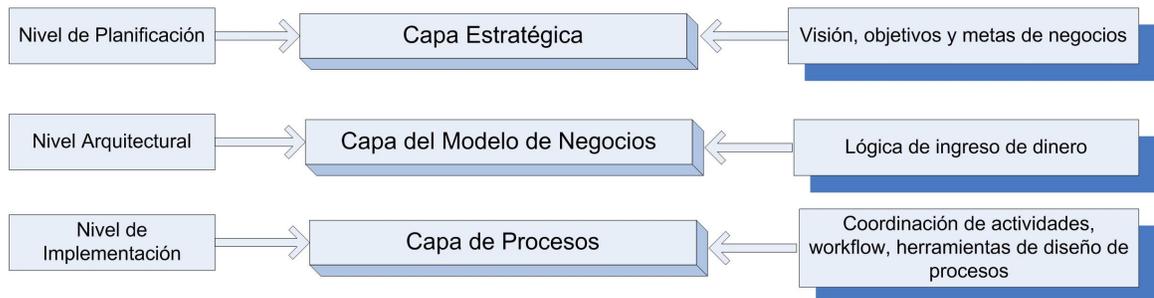


Figura 2.1: Capas en el modelo de negocios

En la capa estratégica se estudian requerimientos relacionados con la planificación del negocio, tales como la visión y las metas, se examinan debilidades y fortalezas internas y se compara con amenazas y oportunidades externas. En este nivel se evalúa la posición de la organización en el mercado y se pactan las estrategias sobre las cuales se fundamenta la elaboración del modelo de negocios.

Esto habilita un nivel arquitectural que consiste en la representación conceptual de las transacciones, flujos de información y redes de asociados, destacando la lógica de ingreso de dinero. Una vez desarrollado el modelo de negocios, se lo implementa en

alguna de las herramientas de procesos de negocios.

La gestión de los procesos de negocios tiene como meta organizar, gerenciar, analizar y optimizar un proceso de negocios (Solomon y otros, 2010).

Por tanto, se diferencian tres conceptos fundamentales: estrategia, modelo de negocio y modelo de proceso de negocio.

Para comprender el alcance de estrategia de negocios por Internet se plantea la siguiente taxonomía:

- **Cartelera publicitaria:** los comerciantes hacen publicidad por Internet para que los usuarios reconozcan su existencia y puedan ser captados como clientes. El fin es informar la existencia de la empresa sin mostrar información de sus productos, ni proveer servicios adicionales.
- **Folleto:** consiste en un sitio Web con información de la empresa y una presentación limitada de sus productos.
- **Catálogos:** se propone una detallada información de productos, incluyendo sistemas de búsqueda y comparación.
- **Servicio:** se usa el sitio Web para proveer servicios adicionales a clientes, tales como mesas de ayuda, comunidades virtuales, ordenes en línea y pago electrónico.
- **Venta en línea:**
  - **Virtual:** se comercia a través de una estrategia totalmente electrónica y se abandona la presencia física del negocio.
  - **Parcialmente Virtual:** se comercia a través de un sitio Web, pero el cobro es a través de un medio telefónico.

- **Sinergia:** se cuenta con un sitio Web para operaciones de comercio electrónico con existencia en paralelo del negocio físico, de modo que a través de servicios adicionales se mejore la relación con el cliente.

Sobre la base de estas estrategias, el modelo de negocios plantea objetivos a nivel gerencial y describe la lógica del sistema de negocios. A partir de esta lógica se elabora el modelo de proceso.

*Un modelo de proceso de negocios es una sucesión de actividades coordinadas, a ser realizadas por personas o recursos tecnológicos, de modo que transformen una o más entradas en un conjunto definido de resultados, con la consiguiente proposición de valor al cliente (Solomon y Litoiu, 2011; Solomon y otros, 2010).*

## 2.2. Modelo de Proceso de Negocios Electrónicos

Un proceso de negocios incorpora tecnologías informáticas y de comunicaciones como recursos para soporte de la estructura organizacional y desarrollo de estrategias.

De este modo, la Internet se usa como plataforma que vincula clientes, vendedores, proveedores y empleados con el fin de comercializar y acrecentar la productividad del negocio. A través de la plataforma se buscan nuevos canales de compra y venta, cadenas de suministros, logística, sistemas de cobro electrónico y servicios post venta (Huang y otros, 2011).

Por tanto, los procesos que posibilitan la lógica de ingreso de dinero pueden ser diseñados para una implementación con recursos informáticos y de comunicaciones, automatizando transacciones intra o inter organización, sobre plataformas informáticas mediadas por redes.

Del análisis de ventajas de negocios electrónicos, se destacan (Huang y otros, 2011):

- Capacidad para ejecutar procesos de negocios con la utilización de tecnologías informáticas y Web.
- Posibilidad de desarrollar plataformas de comercio electrónico, que convocan clientes, proveedores, empleados y asociados por Internet.
- Transformación de la Web en un medio para transacciones de comercio virtual que favorece la venta de productos y la oferta de servicios.

De acuerdo al nivel de incorporación de estas tecnologías se tienen diferentes configuraciones de procesos de negocios electrónicos, tales como comercio electrónico, publicidad, servicios posteriores a la venta, subastas, integración interna de procesos a través de sistemas informáticos y conformación de redes de organizaciones como ventaja competitiva.

Para generalizar se presenta una tipología de procesos de negocios electrónicos (Basu y Muylle, 2011) en diferentes niveles, como se exhibe en la Figura 2.2.

En el nivel 1 se proponen las bases de recursos informáticos y de comunicaciones que sustentan procesos de negocios electrónicos, con el fin de alcanzar metas de disponibilidad y seguridad.

Con el propósito de competitividad se trabaja sobre el nivel 2 para identificar los procesos de negocios electrónicos. Desde la perspectiva de aplicaciones se clasifican tres tipos de procesos: de comercio, soporte de decisión e integración.

El proceso de comercio incluye las aplicaciones que posibilitan las operaciones de compra y venta en línea. Los de soporte de decisión obtienen información y desarrollan modelos analíticos para evaluar capacidades y tomar decisiones, como por ejemplo el uso de herramientas de diseño asistido por computadora cuya sigla en inglés se la

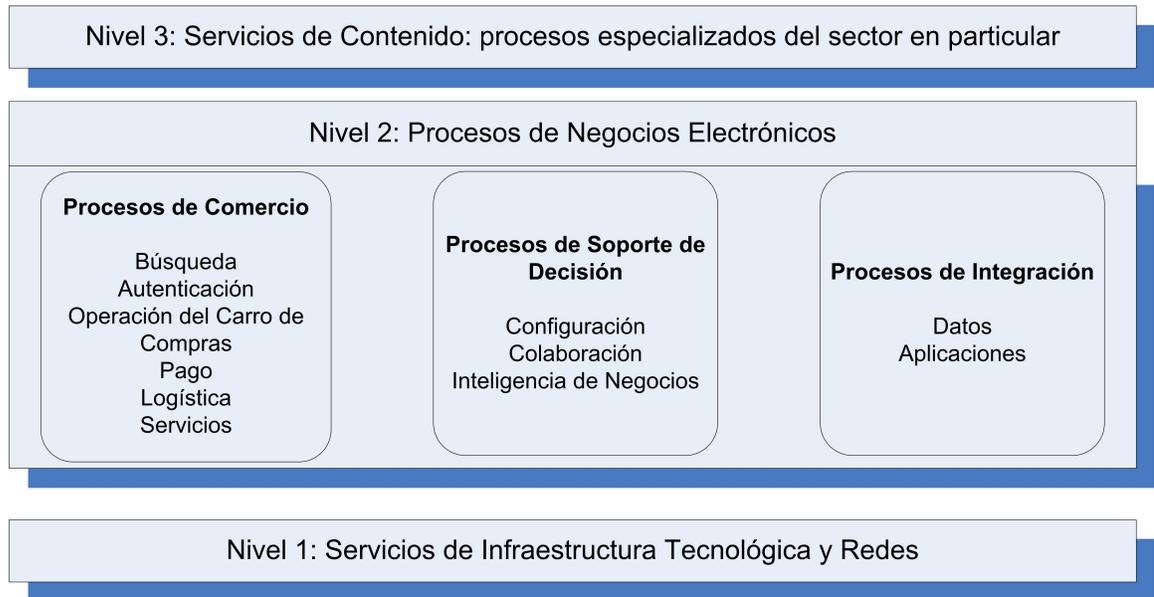


Figura 2.2: Tipos de Procesos Electrónicos de transacciones comerciales

conoce como CAD (Computer-Aided Design) para configuración, colaboración mediante la generación de repositorios de datos compartidos e inteligencia de negocios.

La integración compone sistemas de información para automatizar operaciones entre diferentes elementos y puede ser horizontal entre la firma y sus socios, vertical entre la organización, sus proveedores, distribuidores y clientes, a través de accesos a datos y aplicaciones de asociados. Un ejemplo son los sistemas de información gerenciales ERP, por sus siglas en inglés, Enterprise Resource Planning.

Respecto al nivel 3 procesos de contenidos, se consideran las aplicaciones específicas de una industria o sector que no siguen estándares pero que son necesarias en el desenvolvimiento de operaciones.

De acuerdo a lo descrito se considera la siguiente definición:

*El proceso de comercio electrónico es el desarrollo de operaciones de compra o venta de mercaderías o servicios, sea entre organizaciones públicas o privadas, individuos o el gobierno, conducido sobre redes mediadas por*

*computadoras (The Organisation for Economic Cooperation and Development, OECD).*

Internet es la red de comunicación más reconocida por su llegada masiva a clientes y organizaciones comerciales. El proceso de compra o venta se realiza a través de transacciones entre las partes interesadas.

En función del concepto de transacción electrónica de negocios, la categorización tradicional del modelo de comercio electrónico (Menascé y Almeida, 2000) consiste en:

- **B2C (Business to Consumer):** transacciones entre sitios de negocios y clientes al por menor.
- **B2B (Business to Business):** transacciones entre sitios de negocios al por mayor.
- **C2C (Consumer to Consumer):** subastas entre consumidores finales.
- **G2B (Government to Business) - G2C (Government to Citizens):** operaciones con organismos gubernamentales.

Actualmente, la variedad de operaciones de comercio electrónico produce la incorporación de nuevas categorías ([www.todoecommerce.com](http://www.todoecommerce.com)), entre las cuales se destacan:

- **B2B2C (Business-to-Business-to-Consumer):** transacciones de negocio entre mayoristas que a su vez comercian con clientes minoristas.
- **C2B (Consumer-to-Business):** usuarios comercian con organizaciones.
- **M-commerce (Mobile Commerce):** actividades de comercio electrónico conducidas en ambientes inalámbricos.

- **B2E ( Business-to-Employees)**: transacciones por la cual la organización provee servicios e información a sus empleados.
- **C-commerce (Collaborative Commerce)**: grupos e individuos que se comunican en línea en un ambiente colaborativo.
- **E-learning**: modelos interactivos para la gestión de procesos educativos con el fin de lograr aprendizajes.
- **Bartering**: conocido como trueque, en el cual los interesados intercambian productos y/o servicios.
- **E-trading**: espacio de reunión de compradores y vendedores a través de medios electrónicos con el fin de crear un mercado virtual.

Respecto a la categoría comercio electrónico B2C se plantean las siguientes definiciones (Ho y otros, 2007): (i) Conjunto de operaciones que implican un intercambio de información comercial, con el objetivo de establecer relaciones de negocios e implementarlas por medio de transacciones en redes de telecomunicaciones, (ii) Uso de Internet para compra y venta de productos y servicios, incluyendo el soporte luego de las ventas; (iii) Venta de productos o servicios y su pagos vía líneas telefónicas, redes de computadoras o cualquier otro medio electrónico, sin limitarse a Internet.

Piris y otros (2004) consideran las operaciones de comercio electrónico como vinculación entre tres grupos de redes: un primer grupo privado a nivel organizacional que incluye la gestión interna, información y equipos virtuales, un segundo grupo privado que establece las vinculaciones intraorganizacionales y un grupo público a través de la comunicación global por Internet.

Se observan modelos de comercio electrónico en dos perspectivas, una estrictamente transaccional representada por las funcionalidades del sitio de negocios electrónicos y

vinculadas a la tecnología y otra organizacional, referida a las estrategias de negocios, asociaciones y capacidad de ingreso de dinero.

La debilidad detectada es la disociación entre el modelado de la plataforma tecnológica y el objetivo organizacional, lo cual lleva a perder la visión integral de la empresa y por tanto dificulta la evaluación de rentabilidad.

Por ello, se requiere de una concepción unificada en el diseño de modelos que represente la lógica de ingreso de dinero a través de los procesos de comercio electrónico en función de la tecnología y cuantifique medidas de rentabilidad.

### **2.3. Modelo de Recursos Tecnológicos**

Respecto a las configuraciones de recursos de hardware y software para el diseño de sistemas informáticos se observa un incremento 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 (Fasihul y otros, 2004; Faour y Mansour, 2006).

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 (Andreolini y otros, 2002; Huang y otros, 2009; Pfister, 1996; Ranjan y Knightly, 2008).

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.

En la Figura 2.3 se presenta la arquitectura típica de una plataforma informática de sistema comercio electrónico, cuyo objetivo consiste en alcanzar altos niveles de

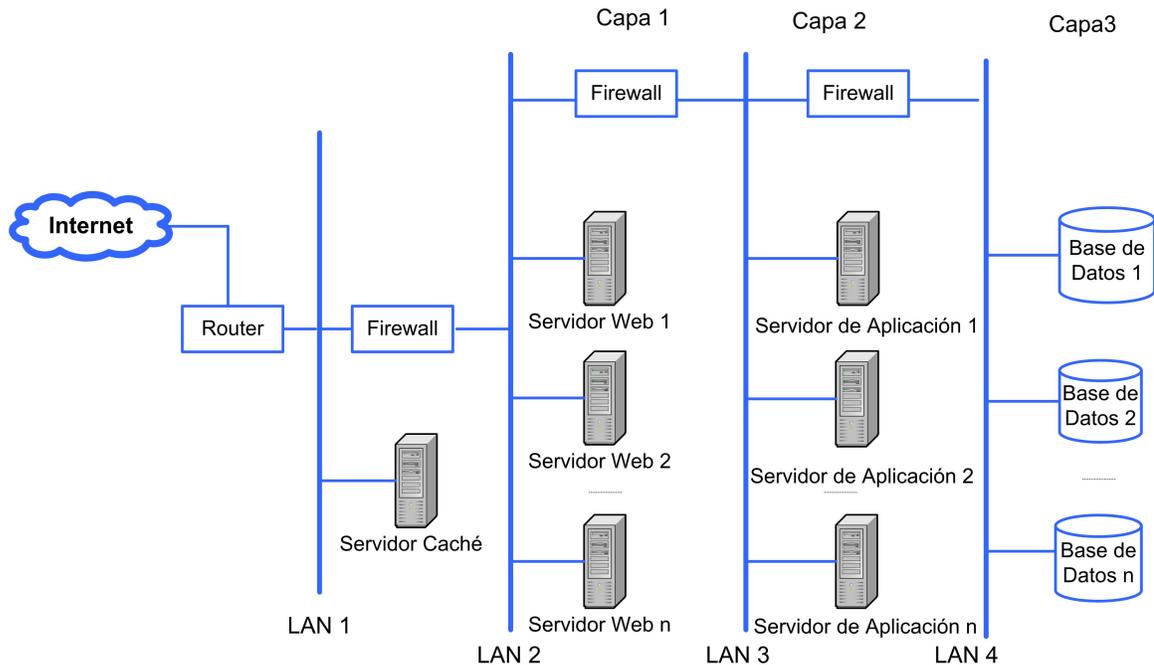


Figura 2.3: Arquitectura multicapa de plataforma informática de comercio electrónico

escalabilidad, disponibilidad y seguridad.

El sistema se conecta a Internet a través de un Router. Los requerimientos pueden ser atendidos por el servidor Caché o ser enviados a la primera capa de servidores Web. La segunda capa se compone de servidores de aplicación y la tercera de servidores de base de datos. Cada capa de servidores es conectada a través de una red LAN, con la existencia de Firewall para seguridad (Menascé y Almeida, 2000).

La arquitectura típica va a ser implementada en diferentes variantes de acuerdo a la configuración de plataforma informática diseñada.

## 2.4. Medición de Desempeño

Previo a la implementación de los negocios electrónicos es conveniente evaluar los costos asociados con aplicaciones, recursos de hardware y cambios organizacionales, en

comparación con los beneficios como retorno de la inversión.

La mejora en el desempeño del negocio está asociada con inversión en tecnología y por tanto con sus costos, beneficios y riesgos. Love y otros (2004) buscan determinar costos, beneficios y riesgos de la inversión en TI (Tecnología Informática) sobre la base de estudios en empresas. Los resultados conducen a la propuesta de un benchmark de métricas para la medición de objetivos e identificación de áreas con necesidad de mejoras. Con respecto a los beneficios concluyen que las empresas analizan sus proyectos de inversión con las dos técnicas tradicionales del cálculo del Valor Neto Actual (VNA) y la Tasa Interna de Retorno (TIR).

El benchmark se compone de tablas de métricas organizadas en beneficios estratégicos, tácticos y operacionales de tecnología informática, sus costos directos e indirectos y factores de riesgo. Entre los principales factores de riesgo se destacan problemas para la adaptación a los cambios, incertidumbre técnica y falta de conocimientos en los sistemas informáticos implementados, pobre mantenimiento y baja seguridad.

Giaglis y otros (1999) proponen un modelo del valor del negocio electrónico, clasificando los beneficios en:

- **Directos:** representados por medidas cuantificables.
- **Intangibles:** atribuidos directamente al negocio electrónico pero difícil de cuantificar, como por ejemplo introducir órdenes de compras en forma electrónica.
- **Indirectos:** medidas cuantificables, pero no atribuidas directamente al negocio electrónico es decir infraestructura del negocio electrónico puede ser utilizada para futuras mejoras en el proceso de negocios.
- **Estratégicos:** impacto positivo medible a largo plazo como resultado de la sinergia entre los recursos implementados, como por ejemplo nuevo posicionamiento

de la organización en el mercado.

Es importante destacar la clasificación de costos como fijos y variables, los cuales se definen en función de las fluctuaciones de una actividad, llamadas generalmente volumen. Este volumen puede medirse en unidades de producto, horas trabajadas, combustible consumido, etc. Si el costo cambia en su valor total en proporción a los cambios en el nivel de una actividad, se dice que es variable. Si el costo permanece invariable en su valor total por un cierto período de tiempo a pesar de los cambios del nivel de actividad, se lo llama fijo (Horngren, 1985).

Gomory y otros (1999) plantean métricas Web con requerimientos de ventas y proponen un modelo multidimensional con información de data warehousing, con una arquitectura organizada en tres ejes fundamentales: las características del tráfico en el servidor Web, las órdenes de productos y las percepciones de los clientes acerca del uso del sitio Web.

Sobre la base del tráfico se construyen las sesiones de clientes y se calculan métricas como tasa de requerimientos, porcentaje de páginas visitadas y cantidad de visitantes. La dimensión órdenes informa las compras de los clientes y su correspondiente valor en dinero y la métrica que la cuantifica es la tasa de conversión o porcentaje de visitantes que ordenan un producto y lo pagan.

También se plantea el tercer eje, la percepción del cliente, medida por la cantidad de clientes que navegando en una página Web acceden a un producto en particular. De estas tres dimensiones surge la métrica llamada tasa de microconversión, que integra medidas de percepción del cliente sobre el producto, incorporación del producto al carro de compras y pago del mismo. La tasa de microconversión provee indicadores para análisis del desempeño del negocio en relación a tecnologías informáticas y de Internet.

Bremser y Chung (2005) centran su estudio de métricas de negocios electrónicos, en la estrategia de la firma y no sólo en la efectividad del sitio Web. Proponen un framework para diseñar métricas sobre la base de la metodología *Balanced Scorecard*, que es un sistema de medición de desempeño centrado en perspectivas financieras, clientes, de los negocios interna y de aprendizaje y crecimiento.

El framework plantea requerimientos claves de medición a nivel táctico y estratégico del negocio electrónico, cuya primera tabla contiene requerimientos de desempeño a nivel de alta gerencia, la segunda, requerimientos a nivel de gerencia táctica y la tercera presenta cuestiones del entorno de negocios electrónicos y busca evaluar fuentes de creación de valor, desarrollos en tecnología, relevancia del modelo de negocios, sistemas de control y riesgos.

Osterwalder y Pigneur (2002) destacan cuatro áreas fundamentales para el diseño de un framework de métricas de modelos de negocios electrónicos: (i) definición del producto y proposición de valor al mercado, (ii) interfase con el cliente, (iii) infraestructura, logística y redes empresariales para la gestión y (iv) aspectos económicos representados por un modelo de ingresos y una estructura de costos.

Respecto al aspecto económico lo consideran como transversal a las tres áreas primeras mencionadas, con el objetivo de identificar la rentabilidad del negocio, rescatando para el modelo de ingresos la habilidad de la firma para lograr beneficios y el modelo de costos como medición de los gastos para crear, vender y entregar valor a sus clientes.

Giaglis y otros (1999) sostienen que la simulación en computadora es una metodología estratégica que puede evaluar las beneficios en relación a los costos en inversiones en comercio electrónico. Por tanto la simulación de modelos de negocios electrónicos permite estimar un conjunto de métricas clasificadas en nivel de negocios y en nivel tecnológico (Menascé y Almeida, 2000). La relación entre ambos brinda un análisis integrado de los negocios en su modalidad electrónica.

## 2.5. Herramientas de Evaluación de Desempeño de Negocios y Tecnología Informática

Mediante la utilización de aplicaciones informáticas, sitios Web o planificación de capacidad se cuenta con herramientas y técnicas para la evaluación del desempeño desde una perspectiva tecnológica en algunos casos y de negocios en otros.

### 2.5.1. Aplicaciones Informáticas

- **Benchmarking:** aplicación que a partir de una prueba configurable permite comparar el desempeño de un sistema respecto a otro. No es una herramienta de planificación de capacidad, por tanto no es utilizada para conjeturar sobre la bondades de un sistema para un uso particular. Se definen en dos perspectivas:
  - **Tecnológica:** método para comparar sistemas informáticos frente a una carga característica de una instalación concreta, efectuándose una comparación con respecto a un prototipo (Puigjaner y otros, 1992).
  - **De negocios:** proceso sistemático y continuo para la evaluación del desempeño organizacional con el fin de medir las estrategias de desarrollo y el crecimiento de la empresa (Stepchenkova y otros, 2010).
- **Monitorización:** herramientas de medición que rastrean el comportamiento de los componentes de un sistema informático para una carga de trabajo dada (Puigjaner y otros, 1992).
- **Testeo de cargas:** proceso de ejecución de un conjunto de scripts que emulan el comportamiento de un cliente en diferentes niveles de cargas y obtienen para

cada una de ellas métricas del sitio (Menascé, 2002).

- **Analytic Hierarchy Process (APH):** método analítico para la toma de decisiones multicriterio con aplicación en problemas de planificación, selección de alternativa óptima, asignación de recursos y resolución de conflictos (Subramanian y Ramanathan, 2012). Se basa en tres principios: descomposición jerárquica, construcción de la matriz de juicio y elaboración de la síntesis de prioridades. La descomposición consiste en la representación del modelo del problema en estudio con una estructura jerárquica, cuyo nivel superior propone el objetivo general, el segundo, los criterios a evaluar para conseguir el objetivo y el último, las alternativas de decisión. En segunda instancia se construye la matriz de juicio que consiste en una comparación de pares de criterios o alternativas respecto al objetivo general. Por último se calculan los pesos por cada alternativa en función de la matriz obtenida y se establece un ranking de prioridades. De este ranking se obtiene la secuencia de alternativas de decisión ordenadas por sus pesos relativos (Lee y Kozar, 2006). Por tanto se pueden evaluar alternativas de decisión en relación con el objetivo general planteado.
- **Key Performance Index (KPI):** establece a priori medidas cuantificables que reflejan factores críticos del éxito de la organización. Estos valores son comparados con indicadores internos o externos, con el fin de medir el nivel de cumplimiento de los objetivos y así se observan y evalúan las operaciones. Básicamente, se definen métricas que permiten medir el nivel de cumplimiento de los objetivos (Tsai y Cheng, 2011).
- **Business Intelligence (BI):** es el proceso de adquisición, interpretación, análisis y utilización de la información en operaciones de negocios, con el fin de comprender el comportamiento interno y externo de la organización. Desde el aspecto interno

se analiza su funcionamiento y desde el externo se toman decisiones de mercado. Ejemplos son sistemas de soporte de decisión y sistemas de información ejecutivos (Chung y Tseng, 2012).

- **Balanced Scorecard.** Es una herramienta propuesta por Kaplan y Norton en 1992, cuya traducción sería Tablero de Comando Balanceado, que a partir de estrategias de negocios propone objetivos factibles de medición. Consiste en un balance entre indicadores desde cuatro perspectivas diferentes (Xiaoping y Chen, 2008): **Financiera:** monitoriza las estrategias de implementación para evaluar cómo se están logrando y mide cómo aparece la empresa frente a los accionistas. **Satisfacción del cliente:** ¿cómo ven los clientes a la empresa?. **Procesos de negocios internos:** evaluar los procesos para detectar cuáles se deben mejorar o profundizar. **Aprendizaje y crecimiento:** ¿qué formación se debe tener para sostener la capacidad de cambio y mejora?
- **Herramientas de Workflow:** TIBCO Business Studio es una herramienta de análisis, diseño, simulación e implementación de procesos de negocios. Las métricas de salida se computan por sesiones de clientes o para cada servicio individual. Las de sesiones analizan los diferentes caminos a seguir para brindar los servicios de comercialización en el sitio Web y las de servicios individuales calculan costos, tiempos, cuellos de botellas o utilización de recursos.

### 2.5.2. Herramientas Web para la Evaluación del Desempeño

- Keynote (<http://www.keynote.com>) es una organización dedicada al testeo y monitorización de aplicaciones Web y móviles. Sus principales productos son la monitorización de aplicaciones Web para la evaluación del desempeño, el testeo de cargas Web y un benchmark tecnológico de negocios electrónicos.

- Transaction Processing Council (TPC) (<http://www.tpc.org>) es un consejo que define benchmark sobre procesamiento de transacciones y base de datos y muestra resultados de evaluación de desempeño.
- WebStone (<http://www.mindcraft.com/webstone/>) es un benchmark de servidores.
- Web Bench: es una herramienta para benchmarking Web y de servidores proxy, con la capacidad de simular requerimientos de múltiples clientes a la vez.
- Standard Performance Evaluation Corporation (SPEC) (<http://www.spec.org>) es una organización sin fines de lucro que mantiene un conjunto de benchmark estandarizados que pueden ser aplicados a sistemas de computación de alto rendimiento.

### 2.5.3. Planificación de la Capacidad

Es un proceso de predicción con el fin de asegurar la disponibilidad de recursos informáticos frente a incrementos en las cargas de trabajos. Este incremento se puede evaluar como respuesta a futuras demandas, de modo que se satisfagan métricas de desempeño y se logre una relación efectividad-costos (Jain, 1991).

Almeida y Menasce (2002) proponen la planificación de capacidad como una herramienta de gestión, la que se sintetiza en la Figura 2.4.

El proceso comienza con el planteo del modelo de negocios y sus objetivos, lo cual dispara el siguiente ciclo de análisis (Almeida, 2002):

1. Identificar metas que representen los objetivos de negocios con el fin de diseñar la arquitectura informática y planificar los requerimientos del sistema. Se con-

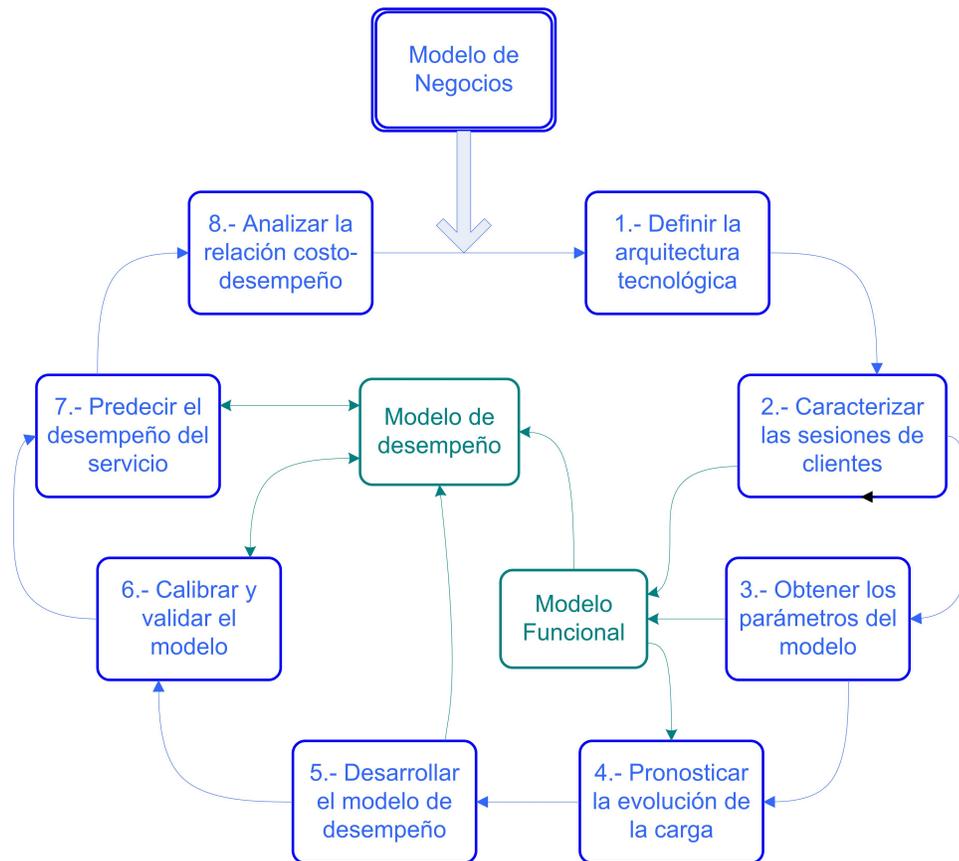


Figura 2.4: Proceso de planificación de capacidad

figuran los recursos de hardware, software y conectividad del sitio de negocios, identificando componentes y servicios.

2. Definir las transacciones electrónicas y su secuenciación como caracterización de las sesiones que un cliente puede adoptar durante su visita al sitio. Ejemplo de sesión es: navegar en el catálogo, buscar un producto, seleccionarlo, agregarlo al carro de compras, registrarse, pagar y salir. Cada transacción se ejecuta en un recurso y cada cliente puede tomar diferentes patrones navegacionales con diferentes frecuencias.
3. Obtener los valores de los parámetros a través de técnicas de monitorización y

benchmarking. En este caso es común utilizar logs de transacciones y accesos a servidores, a partir de los cuales se procesa la información y se computan parámetros de los recursos tecnológicos y probabilidades de transición entre transacciones.

De los pasos 2 y 3 se genera el modelo funcional el cual es una representación de las sesiones que pueden seguir los clientes y la tecnología que va a soportar las transacciones.

4. Pronosticar la intensidad de carga esperada al sitio de negocios para representar futuras frecuencias de arribos de clientes.
5. Plantear el modelo de desempeño con técnicas cuantitativas y modelos analíticos basados en la teoría de redes de colas.
6. Validar las métricas del tiempo de respuesta, velocidad de procesamiento y utilidades de los recursos, calculadas por el modelo en comparación con mediciones de sistemas existentes. Respecto al margen de error aceptable, se considera una exactitud desde el 10 % al 30 %. El proceso de validación interactúa con el modelo de desempeño hasta lograr su calibración.
7. Predecir el comportamiento del sistema evaluando la reacción frente a cambios en niveles de carga y comportamiento de clientes.
8. Analizar las configuraciones obtenidas desde una visión que integre niveles adecuados de las métricas de desempeño y sus costos. Escenarios futuros pueden considerarse para analizar cargas esperadas, costo del sitio y calidad de servicio percibida por el cliente. Este paso indica a la gestión acciones a tener en cuenta para garantizar servicios de tecnología informática y lograr metas de negocios con proyección a futuro.

## 2.6. Framework de Métricas de Desempeño de Modelos de Negocios Electrónicos

Por lo antes expuesto, es necesario disponer de métricas desde una visión integrada de los negocios y la tecnología informática que los soporta para ver cómo influye esta última en los resultados de negocios.

Por esta razón se plantea un Framework de Métricas de Desempeño de Modelos de Negocios Electrónicos, cuya descripción se muestra en la Tabla 2.1. El mismo consta de requerimientos a evaluar, atributos por cada requerimiento y métricas para dichos atributos.

Se observan requerimientos vinculados a la tecnología, tales como sistema informático y redes e Internet, a los clientes con el seguimiento de las funciones que implementan el conjunto de transacciones para sus sesiones, al producto, relacionados con costos y beneficios y por último a la evaluación económica y el análisis financiero.

A partir de estos requerimientos se busca una evaluación económica que fundamente la inversión en el negocio, para luego hacer un análisis financiero del capital invertido.

Respecto a los costos en tecnología informática es importante destacar su clasificación en fijos y variables, considerando a las inversiones como un costo fijo de capacidad.

Para el caso de valores cualitativos puede estimarse una ponderación sobre los ingresos o los costos.

Para realizar el análisis económico se aplica la ecuación del punto de equilibrio. De la comparación de la inversión total con los ingresos netos se calcula el punto de equilibrio (Horngren, 1985), según se muestra en la Ecuación 2.1.

$$\text{Inversiones} = \text{Utilidades} - (\text{Costos Fijos Operativos} + \text{Costos Variables}) \quad (2.1)$$

Requerimiento	Atributo	Métrica
Sistema Informático	Productividad de los recursos de hardware	Velocidad de Procesamiento
		Tiempo de Respuesta
		Utilización de los recursos
	Infraestructura en Servicios Web	Ancho de banda de Internet
		Performance Protocolos Servicios Web
Redes e Internet	Productividad	Transacciones por tiempo
		Tasa de Transferencia
Producto	Proposición de Valor	Utilización de la red
		Valor agregado por T.I.
		Nivel de Innovación
	Calidad	Nivel de Personalización
Clientes	Información estratégica	Índice de Relación precio/calidad
		Comportamiento de Cliente por Sesión
		Calidad de servicio post venta
	Comportamiento en el Sitio	Perfil del cliente
		Cantidad de operaciones por tipo de transacciones realizadas
Costos en T.I.	Costos Variables	Sueldo por cantidad de productos
		Costo de envío de productos
		Sueldo por cantidad de requerimientos
		Costos por cantidad de promociones
	Costos Fijos Operativos	Sueldos para el personal de gestión del Sitio Web
		Sueldos para el personal técnico de mantenimiento del S.I.
		Sueldos de Gestión y Staff para mantenimiento del S.I.
		Costo Proveedor de servicio de Internet
		Hosting del Sitio Web
	Costos Fijos de capacidad (Inversiones)	Adquisición de equipamiento de computación y de comunicaciones
		Adquisición de software de sistema operativo, antivirus y seguridad
		Sueldo por diseño del sistema de comercio electrónico
Actualización de recursos de hardware y software para mejorar capacidad		
Beneficios	Beneficios Directos	Ingresos Totales por Ventas
	Beneficios Intangibles	Nivel de integración de funciones de negocios
		Calidad de Servicios
		Calidad del gerenciamiento a nivel táctico
		Nivel de organización de decisiones no estructuradas
Análisis Económico	Análisis de costo, volumen y utilidad	Calidad de relación con clientes y proveedores
		Punto de equilibrio
		Planificación de utilidades
Análisis Financiero	Retorno de Inversión	Análisis marginales
		Flujos de Fondos
		Valor Neto Actual (VNA)
		Tasa Interna de Retorno (TIR)

Tabla 2.1: Framework de Métricas

En el punto de equilibrio, la rentabilidad del negocio iguala al monto de inversión, por debajo no alcanza a cubrirla y por encima comienza a obtener beneficios económicos.

Cuando el retorno de la inversión se produce en un período de tiempo considerable, se puede actualizar el valor del dinero con las métricas del VAN y la TIR.

## **2.7. Simulación en Procesos de Negocios**

La simulación en el proceso de negocios constituye una estrategia para comprender su esencia, identificar oportunidades de cambio y evaluar el impacto de métricas de desempeño. Además facilita la experimentación de decisiones de mercados con configuraciones alternativas sin necesidad de interrumpir las operaciones corrientes.

La simulación de sistemas informáticos consiste en la descripción del estado de sus componentes para una carga de trabajo variable y aleatoria, por tanto se basa en estados discretos en el tiempo y para trabajar con ellos es adecuado un enfoque de eventos discretos (Madhusudana y Young-Jun, 2005).

Wainer (2003) conceptualiza al paradigma de simulación de evento discreto como cambios de estados del sistema en puntos discretos del tiempo simulado, de modo que se ejecuta una transición de estado frente a la ocurrencia de un evento y se debe tener en cuenta: (i) un conjunto de variables discretas, (ii) un conjunto de eventos a tratar que ocurren en un determinado tiempo de acuerdo a una frecuencia de arribo y (iii) un reloj global que controla el tiempo de simulación.

En un proceso de negocios electrónicos se puede construir un modelo “as-is” para proveer su representación visual, analizar el modelo propuesto e identificar recursos críticos. La simulación de procesos de negocios electrónicos permite trabajar sobre el modelo “to-be” y de este modo plantear escenarios de diferentes diseños de procesos y observar su productividad.

Con la simulación se realizan análisis de sensibilidad sobre modelos, con lo cual se reduce el riesgo de implementar soluciones ineficientes y se prueban conceptos. Es decir,

se comprende el uso de un modelo y se evalúan estrategias de cambio a través de la animación (Greasley, 2000).

## 2.8. Arquitecturas Orientadas a Servicios en los Negocios Electrónicos

En la última década las organizaciones han experimentado un crecimiento en sus plataformas tecnológicas informáticas, que se muestra en la migración de un grupo de servidores a centros de datos con capacidades de alto rendimiento y seguridad. El motivo fue la necesidad de implementar aplicaciones que procesen transacciones de negocios electrónicos, gestionen datos de clientes y soporten herramientas analíticas para la toma de decisiones. Como resultado se puede afirmar productividad en los negocios e incremento de la estructura corporativa (Kaplan y Hughes, 2009).

Otra característica es el nivel de vinculación presentado por modelos de organizaciones con el fin de lograr los objetivos de negocios, que se manifiesta en redes de socios, infraestructura de tecnología informática distribuida, interconexión de empleados, comunicación en línea con el cliente, entre otras (Abuosba y El-Sheikh, 2008).

Plataformas Navegador/Servidor y Middleware son tecnologías de implementación comúnmente utilizadas para soportar aplicaciones Web. Sin embargo, en negocios electrónicos se observa una baja utilización de recursos, indicada entre el 5 % y 20 % de la capacidad total (Castro Leon y Jackson, 2009).

Esto motiva a optar por un cambio en el paradigma de diseño de procesos. Se deja de percibir las aplicaciones y equipamiento como entidades tecnológicas para concebirlas como servicios de negocios. Además las aplicaciones tradicionales están sujetas a una herramienta de programación específica, centradas en servidores que dificultan

invocación distribuida y vinculación dinámica (Mulik y otros, 2008).

Se necesitan unificar soluciones en una red global de recursos informáticos y componer operaciones en un ambiente transparente (Nurcan, 2008), con una arquitectura dinámica y eficiente que integre plataformas heterogéneas con independencia del sistema operativo y del lenguaje de programación. Por ello, se debe pensar en una metodología basada en servicios, que permita vincular aplicaciones en la Web (Mulik y otros, 2008).

Papazoglou (2003) define servicios como aplicaciones autónomas e independientes de una tecnología de software con la finalidad de realizar funciones, que consisten desde simples operaciones hasta composición de aplicaciones distribuidas de complejos procesos.

Una Arquitectura Orientada a Servicios (SOA del inglés Service-Oriented Architecture) es un paradigma para organizar y utilizar capacidades distribuidas que pueden estar bajo el control de diferentes dominios propietarios y es de utilidad para construir aplicaciones que usan servicios disponibles en una red como Internet (Madhusudana y Young-Jun, 2005; Oliver y otros, 2008).

SOA es un estilo arquitectural y su fundamento metodológico es la interacción entre proveedores y consumidores de servicios con pobre acoplamiento e interfaces estándares, cuya finalidad es lograr procesos de negocios ágiles. De sus características se destacan:

- Es un medio de planificación, diseño, implementación y gestión de sistemas de TI.
- Posee pobre acoplamiento, con servicios independientes del sistema operativo o el lenguaje de programación que lo implementa, transparencia respecto a la ubicación, interacción a través de mensajes asincrónicos y descripción estandarizada de interfaces y su semántica.
- Convierte en ágiles a los procesos de negocios a través de la composición de aplicaciones internas o externas a la organización.

El beneficio de la estructura de recursos informáticos en SOA es la concepción de aplicaciones como servicios, su reusabilidad e independencia de una tecnología específica y la capacidad de ejecutar funciones de negocios (Madhusudana y Young-Jun, 2005).

Servicios Web es una tecnología cuya plataforma soporta SOA. La World Wide Web Consortium (W3C) (<http://www.w3.org/>, 2012) los define como:

**“Aplicaciones de software que pueden interactuar unas con otras, cuyas interfaces se basan en XML (eXtensible Markup Lenguaje) y están vinculadas a URIs (Uniform Resource Identifiers). La comunicación se basa en mensajes XML y protocolos de Internet”.**

Así, los servicios Web son un conjunto de aplicaciones o tecnologías con capacidad para operar en la Web e intercambiar mensajes entre sí, con el objetivo de ofrecer servicios. Los proveedores ofrecen sus servicios Web como procedimientos remotos y los usuarios solicitan un servicio Web llamando a estos procedimientos a través de la Web o de una Intranet. La expresión y exposición de estos servicios Web se representan con estándares (Oliver y otros, 2008).

El framework de servicios Web se basa en: SOAP, WSDL y UDDI, que definen la descripción del servicio y la información necesaria para invocarlo. Se considera la representación de datos en el formato eXtensible Markup Language (XML), los estándares de especificación de formatos de mensajes y el mecanismo de invocación llamado Simple Object Access Protocol (SOAP), una gramática de descripción de los servicios Web, sus operaciones, parámetros y valores de retorno llamada Web Service Description Lenguaje (WSDL) y un directorio Universal Description, Discovery and Integration (UDDI) que registra la descripción de los servicios Web y provee mecanismos a los clientes para encontrarlos en la Web (Datla y Goseva-Popstojanova, 2005).

En cuanto a su uso, las tecnologías de servicios Web son una plataforma para el

desarrollo de aplicaciones que describen las funcionalidades de negocios basados en transacciones electrónicas. Con esto se posibilita la integración de negocios a nivel de aplicaciones, utilizando Internet como medio de comunicación global y reemplazando interfaces propietarias y estructuras de datos por estándares (Chen y otros, 2006).

En resumen, las aplicaciones de negocios basados en transacciones electrónicas pueden ser modeladas con SOA e implementadas con tecnologías de servicios Web, lo que posibilita la integración de operaciones de negocios en Internet a través de interfaces y estructuras de datos estándares.

Para comprender la relación entre servicios y operaciones de negocios se plantea la Figura 2.5. Por medio de una acción de servicio, las necesidades de un consumidor se comunican con las capacidades de un proveedor. Esta acción se implementa por transacciones de negocios.

Una transacción de negocios se define como las acciones conjuntas entre dos o más participantes con el fin de intercambiar la propiedad de recursos, teniendo como requerimiento el establecimiento de un contrato entre las mismas. A través de este contrato, dos o más participantes pactan las restricciones y acciones permitidas para la futura operación.

Las transacciones de negocios implementan un proceso de negocios y pueden componerse para formar una coreografía de procesos (MacKenzie y otros, 2006).

Los procesos de negocios requieren la representación de sus interacciones a través de modelos de procesos explícitos, los servicios Web especifican información y requerimientos a nivel de servicios de un proceso pero no tienen capacidades de modelización de procesos y conexiones inter-organizacionales. Esto trae la necesidad de componer servicios Web, por ello surgen los lenguajes de orquestación y coreografía de servicios Web (Guido y otros, 2007).

La orquestación es el proceso de composición de servicios Web sobre la base de una

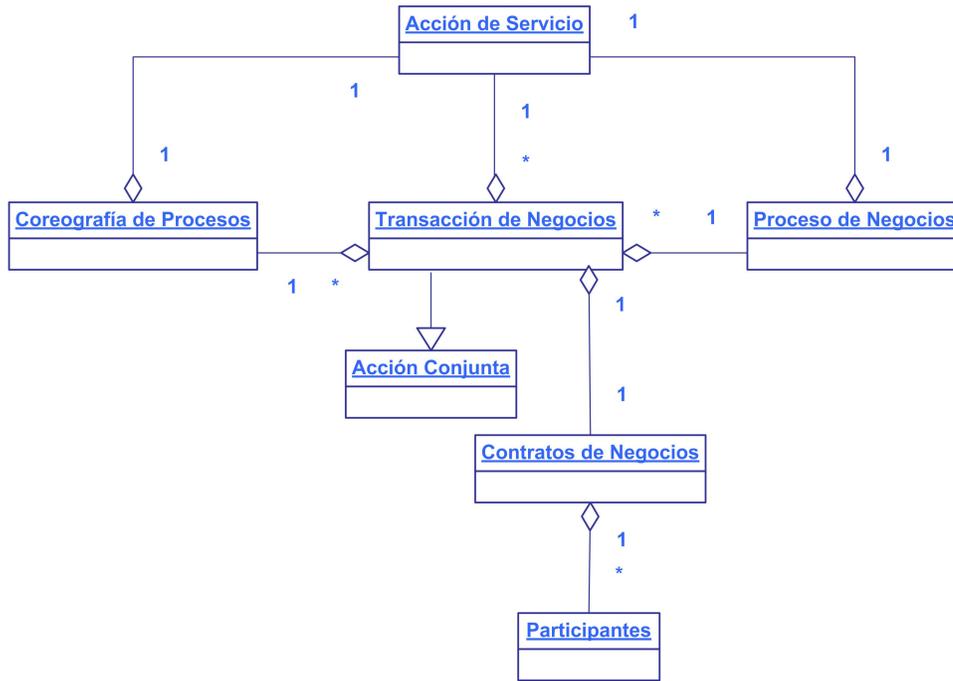


Figura 2.5: Transacciones de negocios en base a servicios

entidad única de control, que define las interacciones y describe la lógica interna de negocios. Esta integración es hecha por un participante y por ello se corresponde con el comportamiento privado de los servicios Web. Mientras que la coreografía esboza la vista global de las interacciones entre un conjunto de servicios para alcanzar una meta común, con aplicaciones que son heterogéneas y de diferentes participantes.

Por tanto, en la orquestación se integran servicios Web por programación de reglas de interacción, que los secuencian, coordinan y gestionan y en la coreografía se define el comportamiento público y observable entre los servicios participantes, sin considerar los detalles privados de su procesamiento y no son ejecutables (Kim y Huemer, 2004).

## 2.9. Escenario de una Arquitectura Orientada a Servicios

SOA se define por la interacción de agentes de software, con un intercambio de mensajes entre tres tipos de participantes: cliente, proveedor y agente de descubrimiento.

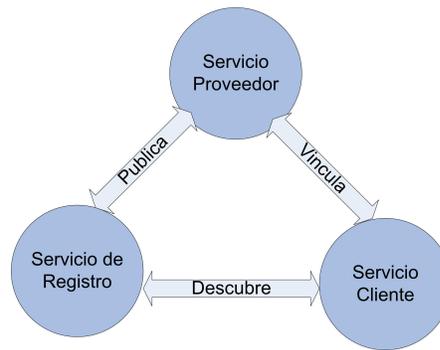


Figura 2.6: Arquitectura básica orientada a servicios

En la Figura 2.6 se observan las siguientes interacciones:

- un proveedor de servicios es responsable de publicar la descripción del mismo en una agencia de descubrimiento.
- el agente de descubrimiento es un servicio de registro que está disponible para ser descubierto por un cliente.
- un cliente que requiere la ejecución de un servicio debe tener la capacidad de encontrar su descripción y de vincularse al proveedor que lo dispone.

Estas interacciones implementadas con tecnologías de servicios Web se representan en la Figura 2.7.

A través del protocolo SOAP se define el mecanismo de intercambio de mensajes XML entre cliente y proveedor. WSDL describe las interfaces y la información necesaria

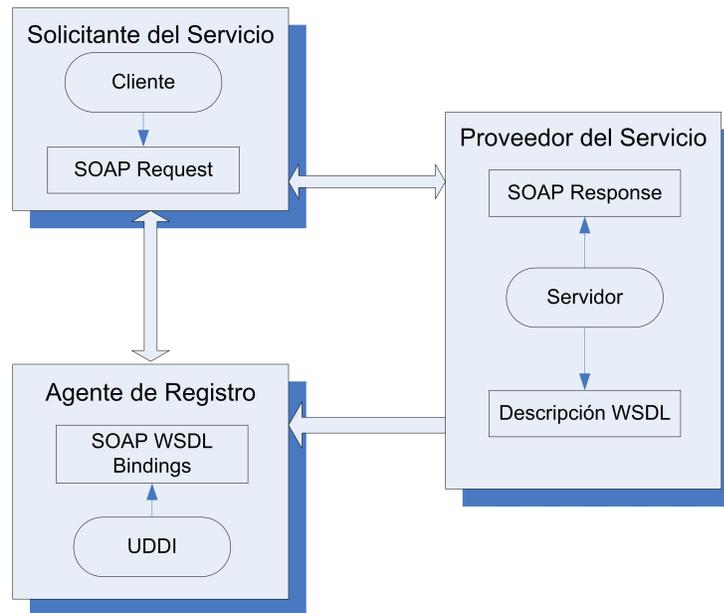


Figura 2.7: Tecnología de servicios Web

para la vinculación con el proveedor de servicios, de modo que el solicitante del servicio las utiliza en sus mensajes SOAP.

UDDI pone a disposición del solicitante un registro de documentos WSDL con servicios que han sido cargados por un proveedor (Dezhgosha y Angara, 2005).

La implementación de servicios Web se basa fundamentalmente en la comunicación cliente y proveedor, de modo que el cliente conoce a priori la ubicación del proveedor. El proveedor de servicio publica un módulo de software accesible en una red y el cliente conoce la información necesaria para vincularse al mismo. Dependiendo de la aplicación se puede incorporar un registro de servicios para la búsqueda en tiempo real de proveedores.

Sobre la base de lo expuesto se presenta en la Figura 2.8 un framework SOA, propuesto por Demirkan y otros (2008).

La capa superior se corresponde al proceso de negocios, la media a servicios y las dos inferiores a los componentes de software y aplicaciones legadas. A partir de los

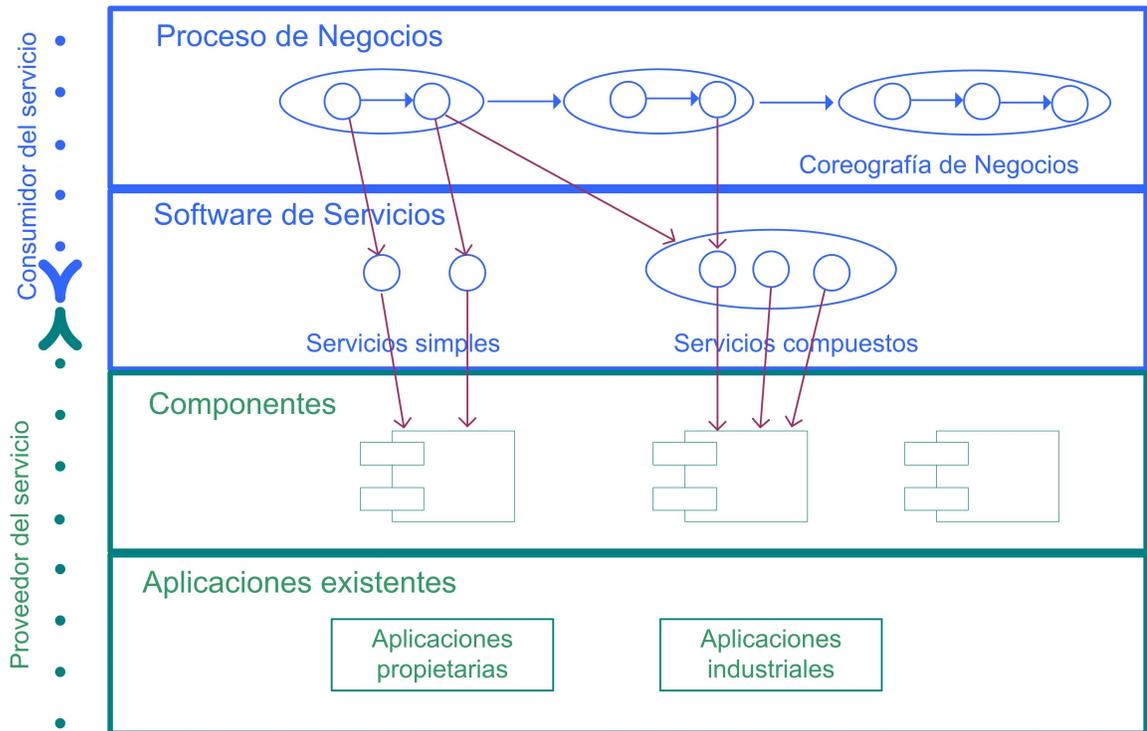


Figura 2.8: Framework de una Arquitectura Orientada a Servicios

componentes de software se crean servicios que pueden ser atómicos o compuestos.

Estos servicios se articulan en una coreografía para llevar adelante un proceso de negocios. Los servicios se componen de una o más transacciones de negocios y pueden ser accedidos por operaciones de negocios sin requerir conocimientos detallados de implementación. Además, se destaca una división entre el consumidor y el proveedor del servicio.

El servicio puede ser provisto por diferentes fuentes, de acuerdo a la estrategia de negocios considerada (Papazoglou, 2003):

- **Diseño, desarrollo e implementación del servicio en la empresa:** una vez especificado el servicio y su interface, el código se elabora dentro de la empresa.
- **Compra, leasing o pago del servicio:** el software de servicio es adquirido

desde un proveedor y no es desarrollado internamente.

- **Tercerización del diseño e implementación del servicio:** una vez especificado el servicio, el desarrollo e implementación de interfaces es tercerizado.
- **Uso de adaptadores para el acceso a software legado.**

## 2.10. Simulación con Herramientas de Workflow

Como primera aproximación a la simulación de sistemas de comercio electrónico se trabaja con una herramienta de Workflow.

Un proceso de negocios puede ser representado por un conjunto de actividades relacionadas con el fin de implementar una operación de negocios. La relación entre actividades es articulada por elementos de control de flujo, quienes determinan el orden de ejecución que puede ser secuencial, paralelo, alternativo o cíclico. Las herramientas de Workflow implementan los procesos desde esta perspectiva (D Ambrogio y otros, 2007).

*Se define Workflow como la automatización de un proceso de negocios, total o en parte, durante la cual documentos, información o tareas se trasladan de un participante a otro por una acción, de acuerdo a un conjunto de reglas de procedimientos (Garcia y otros, 2008).*

Los sistemas de información de Workflow aplican tecnología de la información a problemas de negocios, de modo que automatizan procesos combinando actividades humanas con aplicaciones de TI, con posibilidad de simular su comportamiento (Mulik y otros, 2008; Gang, 2008).

Un metamodelo de Workflow puede organizarse en actividades, condiciones de transición, roles y aplicaciones a invocar, de modo que un grupo de actividades pueden

seguir un flujo secuencial de datos o realizar transiciones de acuerdo a condiciones.

En la Figura 2.9 se presenta un framerwork de modelo de Workflow (Garcia y otros, 2008), cuyas componentes son workflows, procesos y tareas.

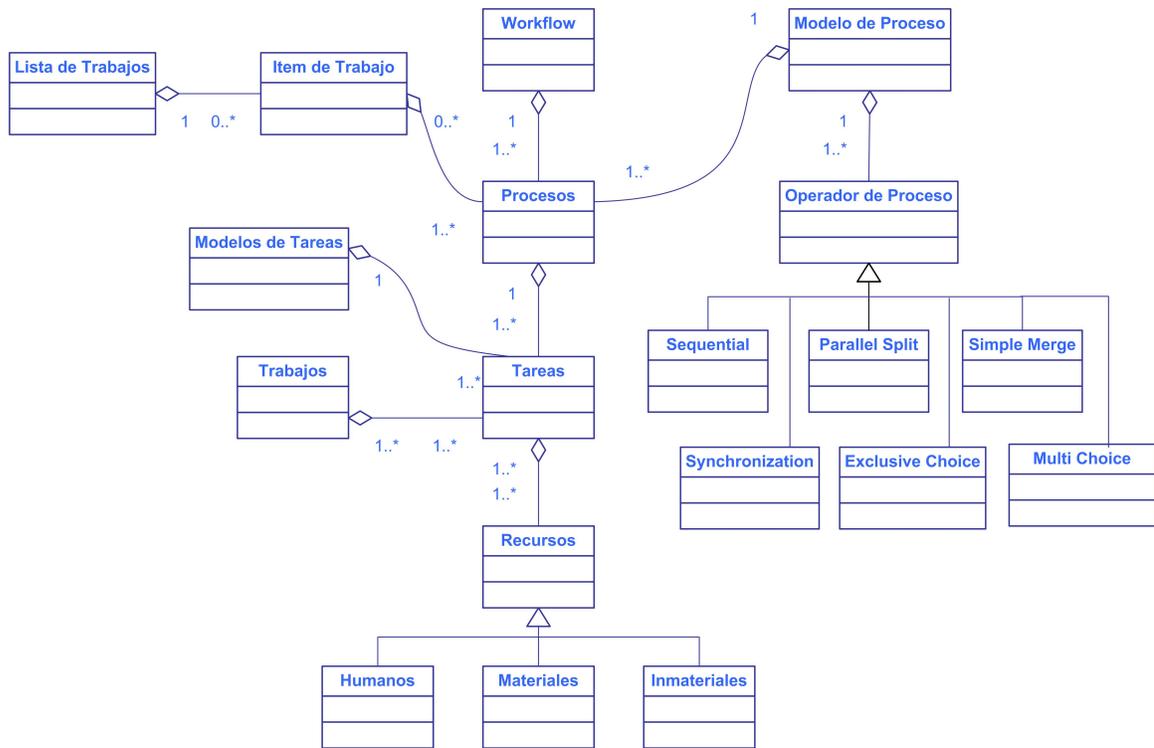


Figura 2.9: Framework de un modelo de Workflow

El modelo de Workflow se forma recursivamente por procesos y tareas. El modelo de proceso indica la estructura de ejecución de tareas y se constituye por operadores que determinan el flujo de datos. Cada tarea es ejecutada por recursos humanos, materiales (equipos, redes, máquinas) e inmateriales (sistemas informáticos, aplicaciones). Estos recursos cumplen con uno o más trabajos. Una tarea es un paso lógico que puede ser ejecutado en diferentes casos que se corresponden con instancias. Un caso en una tarea es llamado ítem de trabajo.

Como se mencionó, un proceso de negocios puede ser visto como un conjunto de actividades ordenadas e interrelacionadas, con el fin de alcanzar una meta organizacional

de negocio (Nurcan, 2008). La interrelación de las actividades se lleva a cabo a través de elementos de control de flujo, que determinan el orden de ejecución secuencial, paralelo o alternativo (Guo y otros, 2008).

Por ejemplo BPEL (Web Service Business Process Execution Language) es un lenguaje de programación basado en Workflow que describe procesos de negocios que orquestan servicios Web (Kim y Huemer, 2004). Estos lenguajes soportan manejo de transacciones de negocios, coordinación del flujo y comunicación con servicios Web.

En BPEL las actividades se comunican por paso de mensajes y pueden ser básicas, como lo son invoke, receive, replay, throw, empty y estructurales como es el caso de sequence, switch, flow para la composición paralela y sincronización y pick para elecciones no determinísticas.

Con una herramienta de Workflow se puede modelar procesos de negocios y analizar su desempeño por simulación de evento discreto (Madhusudana y Young-Jun, 2005). En relación a los modelos de simulación de procesos de comercio electrónico, cada tarea se corresponde con una transacción que puede ser una operación del cliente en el sitio Web, acciones internas de la organización o vinculación con aplicaciones externas. Por tanto, un proceso se forma por una composición de transacciones cuyas funcionalidades obedecen a un servicio de negocio y su análisis de desempeño se implementa por simulación de evento discreto (Madhusudana y Young-Jun, 2005).

Los modelos en tecnologías de Workflow se basan en redes de colas para almacenar ítems de trabajo hasta su ejecución por un participante, ya sea aplicación o recurso humano y de acuerdo al consumo de recursos se calculan métricas de salida.

### 2.10.1. Soporte de Workflow para Diseño de Modelos de Comercio Electrónico

Con la finalidad de construir modelos de comercio electrónico orientados a servicios para simulación con herramientas de Workflow, se elabora un método con base en el framework de la Figura 2.9.

Se deben plantear las metas de negocios y los requerimientos de clientes como base para diseñar la lógica de los procesos.

Cada proceso contiene las operaciones correspondientes a un servicio, las cuales consisten en transacciones de negocios. La vinculación de operaciones entre asociados se orquestan con servicios Web. A cada servicio se le asigna el recurso computacional que lo ejecuta.

Respecto a las metas se tienen en cuenta dos niveles: el tradicional de procesos de negocios, que automatiza las transacciones para lograr las metas organizacionales y el otro comunicacional, correspondiente a las tecnologías de comunicación con el cliente.

Para una mejor comprensión se describen las siguientes etapas:

**Etapas 1:** Caracterizar el negocio a través de la identificación del segmento de clientes, planteo de estrategias de comercialización y definición de metas.

**Etapas 2:** Establecer los servicios a ofrecer y las reglas de ejecución de los mismos. Cada proceso va a incluir los servicios correspondientes a una submeta de negocios.

**Etapas 3:** Elaborar el modelo de transacciones electrónicas. Las transacciones se presentan en un modelo general de estados y transiciones entre estados. Un estado representa una transacción y una transición la probabilidad de avance. Los caminos posibles de un estado inicial a un estado final se corresponden con las sesiones de clientes.

**Etapas 4:** Construir el modelo de Workflow. (i) asignar a cada transacción (tarefas)

los recursos (IT y humanos) y los costos (tiempos); (ii) diseñar el modelo de procesos (lógica de operaciones con los operadores de procesos).

**Etapa 5:** Implementar el modelo en la herramienta de Workflow con posibilidad de ejecutar una simulación. Para este caso se utiliza la herramienta TIBCO. En la Figura 2.10 se observa las objetos gráficos para el diseño del modelo. Los procesos son organizados en pool y lane. El modelo de proceso se estructura en un flujo de información entre operaciones, que puede ser secuencial, con alternativas o ciclos. Cada lane implementa una submeta de negocios.

Operaciones (Tasks)		Eventos		Procesos		Exclusive Gateway	Conectores para secuencia		
									
		Start	End	Pool	Lane		Unconditional	Conditional	Default

Figura 2.10: Objetos para el diseño del modelo

En la Figura 2.11(a) se observa un modelo que comienza con un evento, continúa en forma secuencial con una operación, para luego tener que bifurcarse en un *exclusive gateway* con 50 % de probabilidad. El modelo finaliza con el *end event*. La Figura 2.11(b) muestra un modelo que representa un ciclo, con una probabilidad del 60 % de retornar a la transacción *Browse* y del 40 % de continuar a la transacción *Search*.

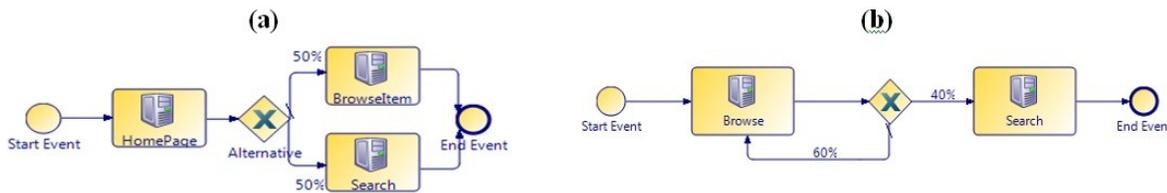


Figura 2.11: (a) Estructura alternativa. (b) Estructura cíclica

**Etapa 6:** Parametrizar la simulación. Cada actividad tiene asociada una demanda, que es el requerimiento de servicio promedio por trabajo en el recurso (IT o humano)

para ejecutar una transacción. Las entradas son el número de casos y la frecuencia de arribos.

**Etapas 7:** Plantear métricas de performance. Las métricas se plantean en función del número de casos, tiempo de simulación y recursos consumidos, se obtienen por servicio y por sesión de clientes. Se puede mencionar: tiempo promedio de un caso en el sistema, cantidad de casos por servicio, cantidad de casos por sesión de cliente, utilización de los servicios (cuellos de botellas y subocupaciones), entre otras.

### 2.10.2. Modelo de Workflow del Proceso de Comercio Electrónico B2C

Para comprender el método propuesto se propone un modelo de comercio electrónico B2C orientado a servicios. El diseño del modelo y sus parámetros se basan en el caso de estudio del Capítulo 4 de Menascé y Almeida (2000) y contiene los siguientes componentes:

- **Cliente:** el cliente hace requerimientos al sitio de negocios desde un browser de Internet y espera la respuesta a través del mismo.
- **Internet:** los requerimientos del cliente son enviados al sitio a través de Internet y recibidos por la misma red.
- **FrontEnd del servidor:** como componentes frontales del servidor se cuenta con un router, un servidor de caché y redes locales que los comunican.
- **Sitio Web:** conformado por las transacciones visitar la página principal, buscar un producto, agregar al carro de compras, registro del cliente, cobro del producto con tarjeta de crédito y procesar la orden.

- **Servicios Web:** la búsqueda de un producto y el cobro a través de la tarjeta de crédito se implementa con tecnologías de servicios Web.

El modelo implementado en la herramienta TIBCO se muestra en la Figura 2.12, donde se observa su arquitectura, conformada por un proceso general presentado en el pool `EcommerceWebServicesArchitecture`, subdividido en lanes:

- **Website:** servicios para la gestión interna de la lógica del negocio.
- **WebServiceSearch:** servicio de búsqueda.
- **WebServiceCredit:** servicio de evaluación del estado del crédito del cliente.
- **FrontEnd:** servicios de conexión del sitio de negocios con Internet.
- **WAN:** comunicación en Internet.
- **Client:** proceso del lado cliente.

La transacción se parametriza con distribuciones de probabilidad constante del tiempo de servicio en el recurso informático. El flujo de información sigue una estructura secuencial, de decisión o ciclos. Los casos se generan en el evento de entrada con una distribución de probabilidad constante del tiempo entre arribos.

Cada caso comienza con el cliente, el cual realiza su solicitud desde su computadora e ingresa al sitio de negocios a través de Internet. El proceso `FrontEnd` contiene un servicio de Caché que puede atender el requerimiento o dirigirlo al `HomePage`. Las operaciones de búsqueda y evaluación del crédito del cliente se hacen a través de servicios Web. El modelado de servicios Web consiste en el envío del requerimiento por parte del solicitante, la atención por parte del proveedor y la correspondiente respuesta.

Para analizar el comportamiento del cliente en el modelo se miden las siguientes sesiones:

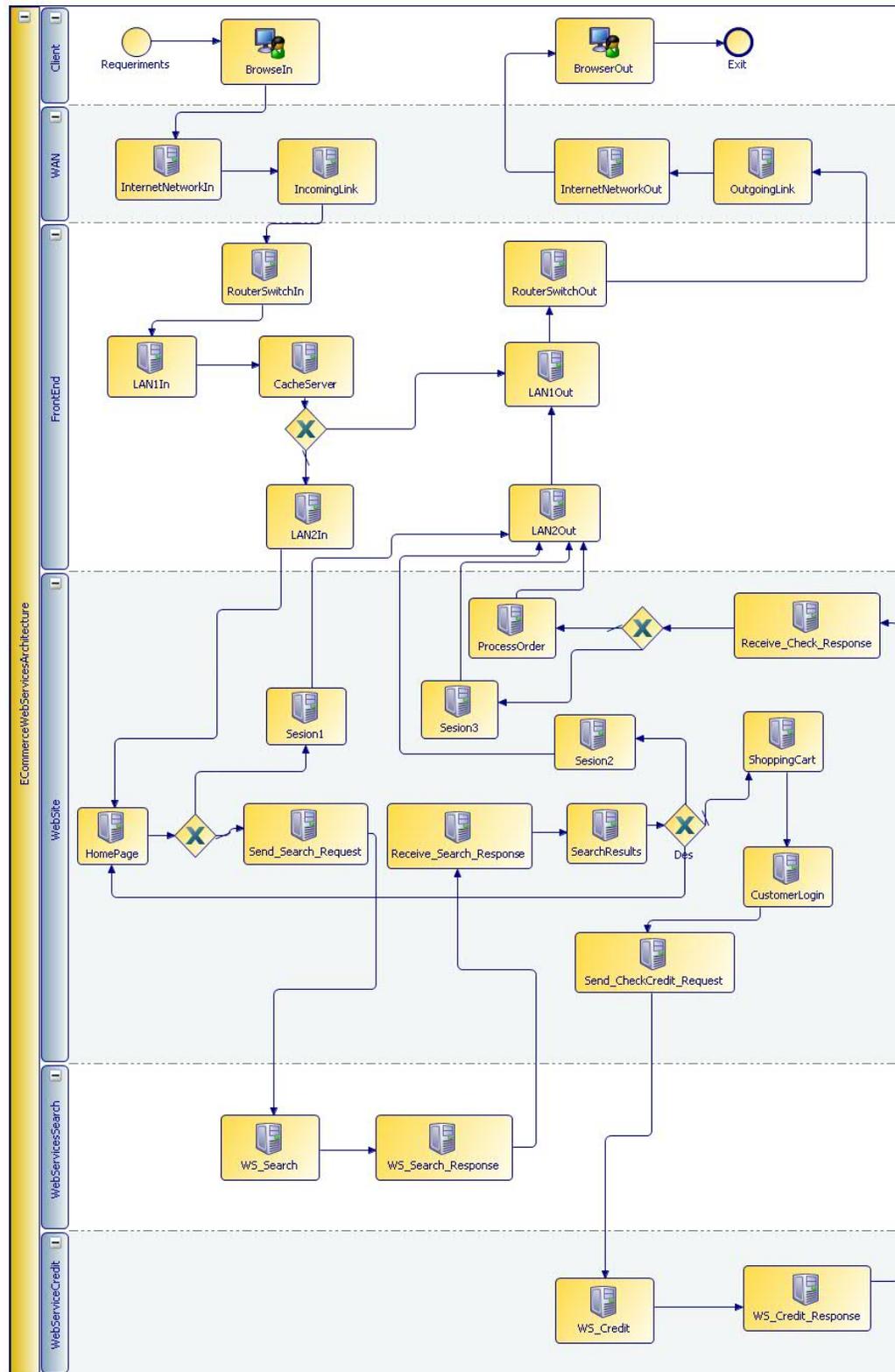


Figura 2.12: Modelo de comercio electrónico B2C

- **Sesión 1, Homepage:** cliente que sólo visitó la página inicial y abandonó el sitio.
- **Sesión 2, Search:** cliente que hizo búsquedas.
- **Sesión 3, Without Credit:** cliente que no pudo efectuar la compra por no tener autorización del crédito.
- **Sesión 4, Order:** cliente que compró.

### Simulación y análisis de resultados

En el primer caso de simulación se analiza el comportamiento del proceso para diferentes cantidades de clientes en el sitio. El escenario consiste en la variación del número de clientes, con un tiempo entre arribos constante de 0,5 segundos. El número de casos comienza en cien y finaliza en mil, con incrementos de cien.

En la Figura 2.13 se observa el tiempo promedio de un caso en el sistema que comienza en 0,012 minutos, para luego estabilizarse en 0,0121 minutos desde los 400 casos. Se puede concluir que el proceso simulado cuenta con una arquitectura eficiente para una frecuencia de arribos de clientes de 0,5 segundos.

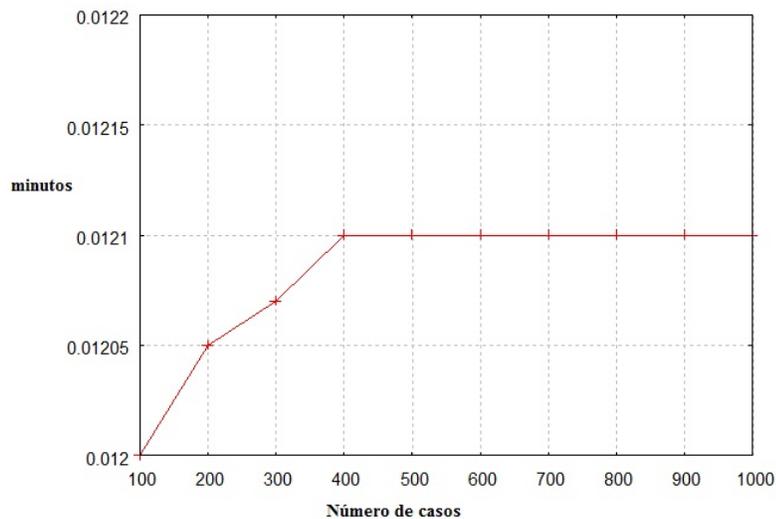


Figura 2.13: Tiempo promedio de un caso en el proceso

En la Figura 2.14 se muestra la cantidad de casos por tipo de sesión, lo que posibilita la evaluación del cliente en los diferentes servicios y sobre todo estimar un ingreso promedio en función de los clientes que compran.

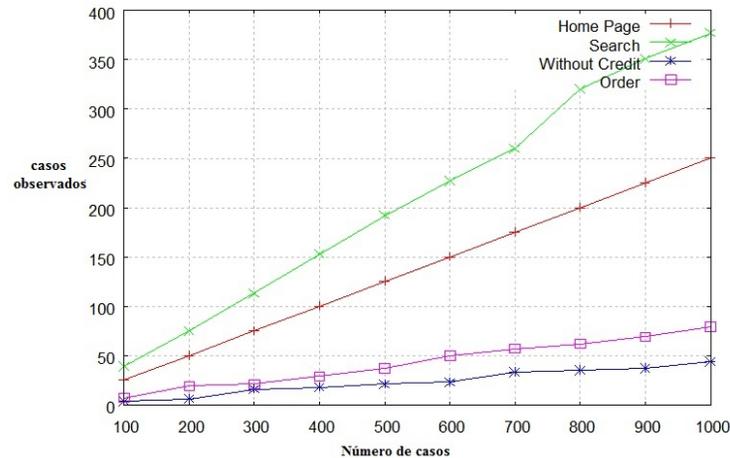


Figura 2.14: Número de casos por tipo de sesión

En un segundo escenario se simula el comportamiento del proceso frente a una variación del tiempo entre arribos. Para ello se dejan constante el número de casos en 500 y se modifica los tiempos entre arribos.

En la Figura 2.15 se muestran los tiempos promedio de un caso y se observa una saturación del proceso para frecuencias inferiores a 0,14 seg, esto significa que la plataforma informática ha alcanzado una utilización al 100 % de los recursos y por tanto los nuevos requerimientos quedan esperando en cola sin ser atendidos y los tiempos de respuesta crecen exponencialmente.

Las utilizaciones de los servicios muestran un 70 % para el link de entrada y un 98,85 % para el link de salida (Figura 2.16). De la observación de las utilizaciones se determinan los recursos críticos del sistema y de este modo se pueden proponer nuevas configuraciones como escenarios de futuras simulaciones.

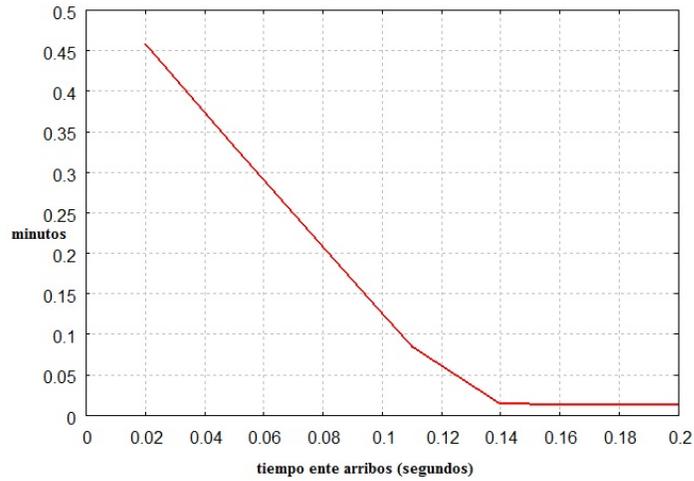


Figura 2.15: Tiempo promedio de un caso para diferentes frecuencias de arribos

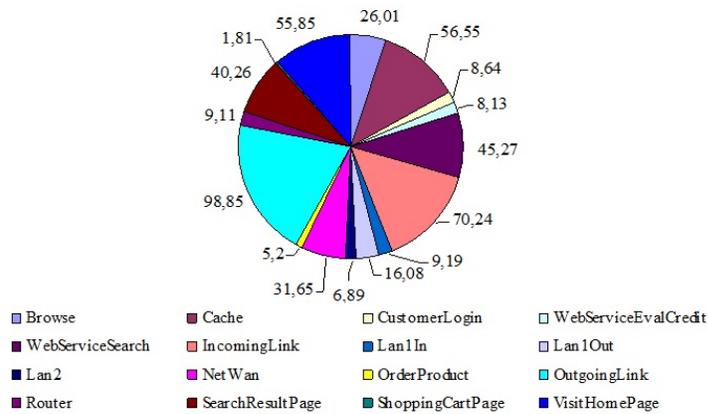


Figura 2.16: Porcentajes de utilización por servicio

De la ejecución de la simulación se encuentran métricas tecnológicas como el tiempo de respuesta de una sesión de cliente y utilización de los recursos, así como de negocios para el caso de la cantidad de clientes que visitan la página principal del sitio y lo abandonan, sólo realizan búsquedas o confirman la orden y ejecutan la compra.

## 2.11. Arquitectura de Referencia del Modelo de Negocios Electrónicos

Con base en lo antes expuesto se comienza con la propuesta de la metodología de modelización para simulación orientada a servicios de transacciones de comercio electrónico, la cual requiere de un framework que fundamente las diferentes etapas y sus alcances y sea capaz de:

- Plantear una abstracción a nivel de negocios de la cual se infiera las transacciones que implementan las estrategias comerciales.
- Definir los recursos de hardware y software que configuren la plataforma informática del proceso de comercio electrónico.
- Describir el patrón de comportamiento de clientes en su navegación en el sitio Web.
- Obtener métricas de desempeño que presenten una visión integrada de indicadores tecnológicos y de negocios.

En la Figura 1.4 del Capítulo 1, se planteó el modelo de referencia de negocios electrónicos propuesto por los autores Menascé y Almeida, en el cual se destaca su organización en cuatro bloques divididos en dos grupos. El grupo superior se concentra en el modelo de negocios y las transacciones que implementan el proceso de negocios, mientras que el inferior describe el comportamiento de clientes y la plataforma tecnológica que soporta el sitio.

Otro modelo de referencia es el propuesto por Schmid y Lindemann (1998), en el cual se definen cuatro vistas, como se muestra en la Figura 2.17. La vista superior se

corresponde con el modelo de negocios de acuerdo a la estrategia organizacional, en un nivel medio se identifican los servicios sobre los cuales se planifican las transacciones del sitio y por último se propone la infraestructura tecnológica que soporta las operaciones de comercio electrónico.

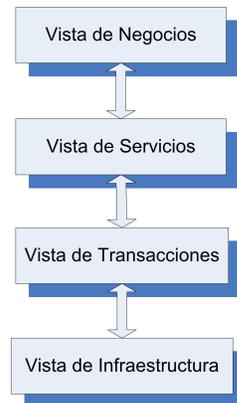


Figura 2.17: Vistas del Modelo de Comercio Electrónico

De acuerdo a una concepción orientada a servicios se mostró un framework propuesto por Demirkan y otros (2008), representado en la Figura 2.8. En el mismo se destacan el diseño de una coreografía de servicios en el proceso de negocios, la identificación de los recursos de software que implementan los servicios y la configuración de los componentes de hardware que ejecutan el software. Se distinguen dos niveles, uno del consumidor de servicios en el proceso de negocios y otro del proveedor para los componentes de hardware y software que los ofrecen.

Por tanto, el proceso de negocios electrónicos requiere una arquitectura de referencia que contemple por un lado, la estructura funcional y sus servicios a nivel de negocios y por otro la infraestructura tecnológica que define los componentes de hardware y software a nivel de recursos. De este modo se puede analizar la forma en que interactúan ambos niveles y evaluar el desempeño de las transacciones que implementan las operaciones de comercio electrónico, en relación con su capacidad de producir ingresos

económicos.

De acuerdo a lo antes expuesto se define la Arquitectura de Referencia Orientada a Servicios y se la detalla en la Figura 2.18.

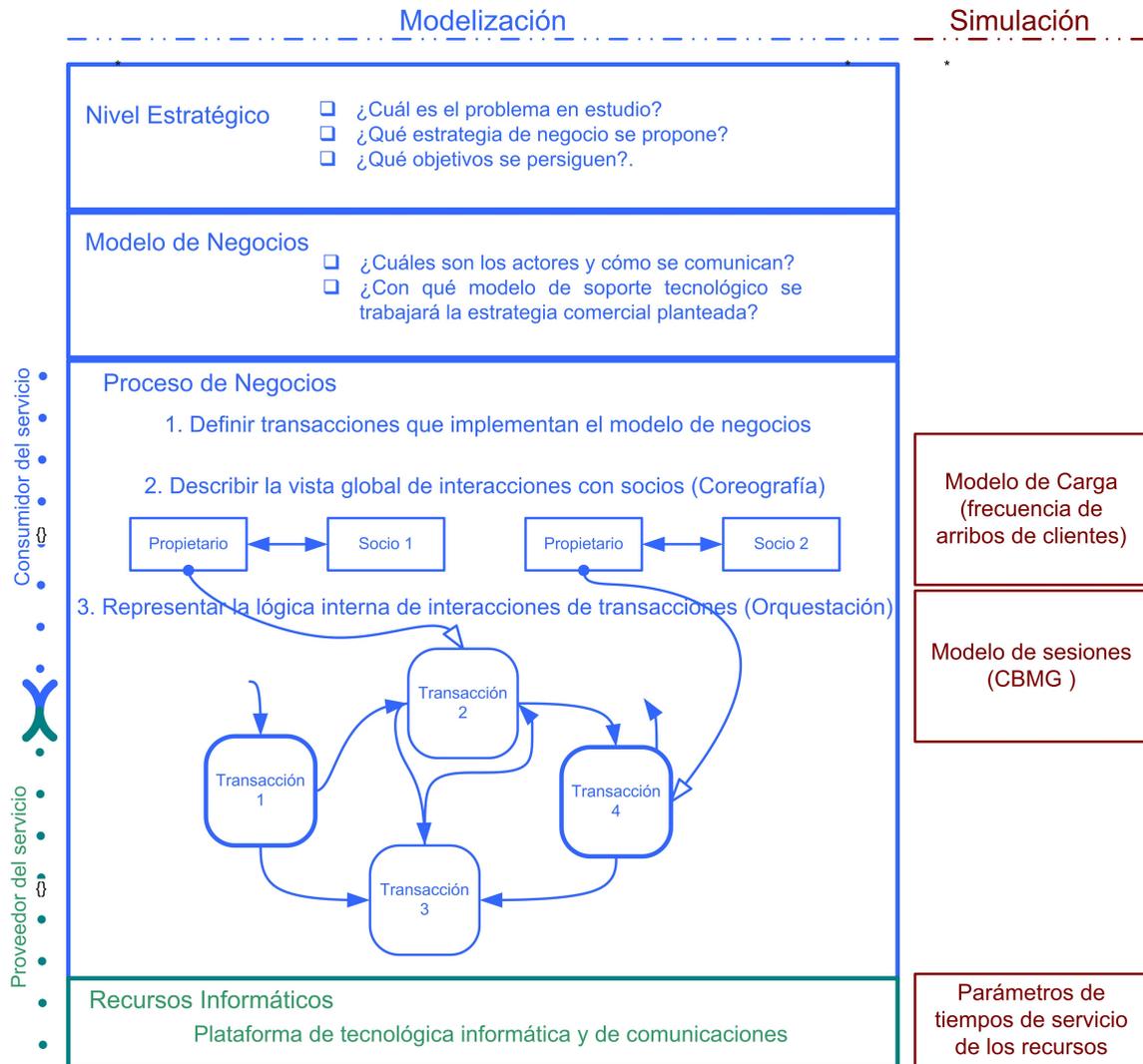


Figura 2.18: Arquitectura de Modelización y Simulación Orientado a Servicios

En la arquitectura se observan dos partes básicas, una correspondiente a la modelización y otra a la simulación. La modelización consta de cuatro niveles, cuya correlación desde el superior al inferior conlleva a la construcción del modelo.

En el nivel estratégico se define el problema en estudio, lo cual consiste en planificar el negocio, haciendo explícita la visión que se tiene sobre el mismo, determinando su tipo y concretando los objetivos y metas específicas.

Sobre esta base se elabora el modelo de negocios, que consiste en las siguientes actividades:

- Planteo y discusión de las políticas organizacionales.
- Especificación del producto que se va a comerciar o del servicio que se va a ofrecer.
- Conformación de la estructura organizacional y definición de los actores intervinientes en la red de negocios.
- Especificación de los roles para cada uno de los actores.
- Exploración de las oportunidades de negocios como medio para puntualizar y evaluar la lógica de ingreso de dinero.

Para el modelo de negocios se requiere un proceso afín que describa la vista global de operaciones entre actores intervinientes, identifique las transacciones que implementen las operaciones del sitio electrónico y establezca la lógica interna de interacciones.

En el nivel de procesos de negocios se requiere de herramientas que modelen los servicios, sus transacciones y las interacciones. Para ello, se plantea una coreografía que describe una vista global de interacciones con asociados y una orquestación que vincula las transacciones internas que ejecutan los servicios. La implementación de estas transacciones puede ser en parte o totalmente con servicios Web.

Una vez definidos los servicios, sus transacciones y la composición, se puede iniciar el último nivel, que por su ubicación inferior quiere significar la plataforma informática y de comunicaciones que reproducen el soporte tecnológico del proceso de negocios.

Respecto a los recursos informáticos, se especifica el equipamiento informático y de comunicaciones, aplicaciones, sistemas operativos de red, software de seguridad y antivirus. Además se debe distinguir si los recursos son propios, tercerizados o delegados.

Una vez construido el modelo, éste se completa con los parámetros que se requieren para la simulación. Tomando como base el nivel de proceso de negocios, se necesita trabajar en dos aspectos el modelo de carga y el modelo de sesiones, ambos vinculados al cliente.

Para el modelo de carga de trabajo se analiza la frecuencia de llegadas de clientes para encontrar la distribución de probabilidad que ajuste a los datos y así calcular el tiempo entre arribos al sitio Web.

Por otro lado para el modelo de sesiones de los clientes se propone utilizar el diagrama CBMG (Customer Behavior Model Graph) para diseñar la integración de transacciones e indicar las diferentes acciones de las sesiones como transiciones de estado con sus probabilidades de ocurrencia.

La información se obtiene con monitorizaciones del sitio Web, sobre las cuales se establecen las frecuencias de acceso a las diferentes transacciones y de este modo se calculan las probabilidades.

Los recursos computacionales tienen como parámetro el tiempo medio de servicio de cada transacción procesada. Esta información se obtiene a través de las monitorizaciones y del tratamiento de los datos recolectados.

En la arquitectura se puede observar un proveedor de servicios indicado a nivel de recursos y de transacciones del proceso de negocios y uno de consumo de servicios indicado por la vista global de interacciones de servicios. De este modo se destaca una división entre el proveedor y el consumidor de servicios, tal que existen servicios publicados, los cuales pueden ser accedidos asincrónicamente a través de los mensajes de los consumidores.

## 2.12. Etapas de la Metodología de Modelado y Simulación

Las operaciones de comercio electrónico deben reaccionar a requerimientos de clientes no previstos a priori y cuya aparición no sigue un comportamiento determinístico. Además, deben tener capacidad para responder frente a las presiones de sus competidores y a los cambios tecnológicos. Por tanto, requieren estar preparadas para atender solicitudes en forma aleatoria y reestructurarse con agilidad.

En este contexto, se enfrentan a eventos asincrónicos de negocios, como por ejemplo el arribo de una consulta de datos sobre productos, una solicitud de compra de un cliente o un requerimiento de pago.

Esto significa que los eventos no siguen una secuencia determinada sino más bien una ejecución dinámica y con entradas que se pueden dar en paralelo y asincrónicamente.

Teniendo en cuenta la característica de orientación a eventos y la arquitectura de referencia expuesta en la Figura 2.18 para soporte de aplicaciones orientadas a servicios, se elabora la metodología de modelización para la simulación.

En la Figura 2.19 se presentan las etapas de la metodología propuesta. Las etapas orientan la elaboración del modelo de simulación desde el planteo del problema, su construcción en el formalismo DEVS y su simulación.

Como es orientada a servicios, se plantea una etapa de proceso de negocios como composición de servicios. Dicha composición radica en la coreografía entre servicios de socios de negocios y la orquestación de servicios propios.

Una vez finalizada las etapas correspondientes a la sección de negocios se propone una configuración de plataforma informática y de comunicaciones que soporte el proceso de comercio electrónico.

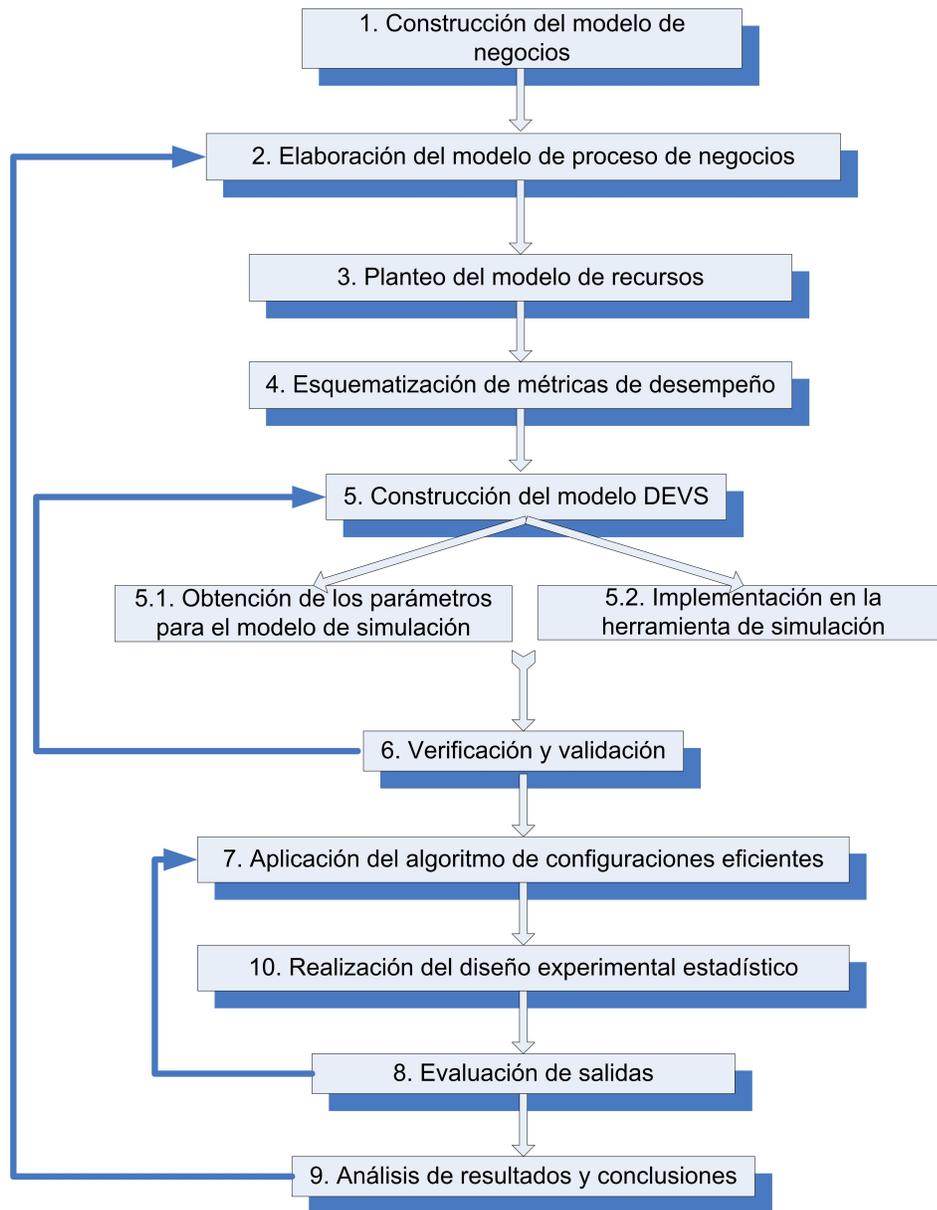


Figura 2.19: Etapas de la Metodología de Modelización y Simulación

Antes del diseño del modelo DEVS se necesitan métricas de salidas para la evaluación integrada de tecnología y negocios.

Una vez construido el modelo DEVS se procede a su verificación y validación. Con la verificación se muestra el correcto diseño del modelo y con la validación su capacidad de representación de la realidad.

A partir de las métricas obtenidas se requiere la interpretación de las mismas, por ello se plantean indicadores como referencia de comparación y un algoritmo de configuraciones eficientes como técnica para determinar los valores que los alcanzan.

Con la configuración adecuada del modelo se procede a evaluar la incidencia de las variables de entrada sobre las de respuesta. Por ello se propone el diseño experimental estadístico como estrategia. Por último se evalúan las salidas, se presentan resultados y se elaboran las conclusiones. En algunas etapas se pueden producir ciclos retornando a las anteriores para reconstruir el modelo de acuerdo a observaciones y modificaciones necesarias.

En las siguientes secciones se describen las etapas de la metodología.

## 2.13. Construcción del Modelo de Negocios

Se inicia la construcción del modelo a nivel de negocios con la definición, caracterización del problema, evaluación de las estrategias y el planteo de los objetivos de negocios.

De la exploración de las organizaciones existentes de comercio electrónico se categorizan los siguientes segmentos de negocios:

- Ventas de productos sin ser fabricantes.
- Fabricación o elaboración y ventas de sus propios productos.
- Ventas de productos configurables por los clientes.
- Relaciones de negocios entre usuarios particulares.
- Venta de servicios.
- Marketing a través de redes sociales.

Para los segmentos de negocios se cuenta con estrategias comerciales mediante el uso de recursos informáticos para el planteo del modelo de negocios, las cuales fueron descriptas en la taxonomía de estrategias de negocios con Internet de la Sección 2.1.

## 2.14. Elaboración del Modelo de Proceso de Negocios

Definidos el modelo y la estrategia de negocios se diseña el proceso de negocios, lo cual consiste en proponer las transacciones que implementan la estrategia comercial, elaborar la coreografía de interacciones con socios y orquestar las transacciones internas. La coreografía se diseña con el diagrama de interacciones UML y la orquestación con el gráfico CBMG.

Del modelo y la estrategia de negocios se decide cuáles son los socios con los que se va a interactuar y se acuerdan las transacciones de negocios del proceso de comercio electrónico.

Con los socios y sus roles se elabora una red global que muestra las interacciones entre las operaciones de negocios, como por ejemplo la integración con el servicio de cobro electrónico realizado por un tercero como servicio Web.

Con las transacciones se construye un modelo CBMG que representa las sesiones para un tipo de clientes navegando en el sitio. Este gráfico indica la transición entre los estados o funciones del sitio con una probabilidad de ocurrencia, es decir, si un cliente está en un estado *Navegar*, tiene una probabilidad de seguir con la transacción *SeleccionarProducto* y otra para *Salir*.

Se presenta un caso de venta en línea de libros para ejemplificar los conceptos planteados, en el cual se debe modelar una consulta de los datos de un producto al inventario del proveedor. El propietario cuenta con un sitio de negocios que consulta el inventario del proveedor con tecnologías de servicios Web.

### 2.14.1. Coreografía de las Interacciones con Socios

En la Figura 2.20 se muestra la coreografía del ejemplo, que establece la interacción entre el cliente, el propietario del sitio de ventas y el proveedor en su rol de socio.



Figura 2.20: Coreografía del servicio de consulta

El cliente solicita datos de un libro al sitio Web, el cual se comunica con el proveedor requiriendo la información necesaria. El proveedor envía la respuesta al sitio de ventas para ser devuelta al cliente.

### 2.14.2. Orquestación de Servicios Propios

La orquestación de los servicios se implementa con el diagrama CBMG y su construcción puede ser realizada de acuerdo a dos posibilidades:

- **El sitio Web está implementado:** se recorre el sitio, se obtienen las principales transacciones y se elabora una tabla de doble entrada. La tabla contiene en las filas y columnas cada una de las transacciones, significando la transacción en la fila el punto origen y la de la columna el destino. Cada celda marcada representa la existencia de una transición entre las transacciones que rotula la fila y la columna. Para encontrar las probabilidades de transición se aplica una herramienta de monitorización en el sitio web que provee las cantidades de clientes que toman cada punto de salida en un tiempo de observación. En forma porcentual se calculan las probabilidades.
- **El sitio Web no está implementado:** se toman modelos CBMG propuestos en la bibliografía o se consulta a expertos para hacer una estimación de probabilidades.

De las dos posibilidades se considera conveniente la primera ya que cada sitio Web tiene sus particularidades para lo cual las sesiones de clientes son diferentes del resto.

La creciente demanda de la compra por comercio electrónico minorista y los avances de la tecnología informática hace que los sitios Web incorporen continuamente nuevas transacciones, de modo que los modelos CBMG cambian continuamente.

Un problema fundamental es que por cuestiones de competencia entre empresas no es siempre posible divulgar este tipo de información ya que en función de la misma se puede conocer su productividad de negocios. Por ello una alternativa es la utilización de diagramas CBMG propuestos por la bibliografía o diseñados a partir de información

de expertos en el tema.

En la Figura 2.21 se muestra un esquema general del modelo CBMG en donde se detallan las transacciones que componen las funciones del sitio de negocios y las transiciones entre ellas.

Las transiciones entre las diferentes transacciones son aleatorias, por tanto de la transacción  $i$  pueden producirse transiciones a las transacciones  $j$  y  $k$  de acuerdo a una probabilidad de ocurrencia, con la consideración de que la suma de sus probabilidades sea igual a uno.

Existe una transición de probabilidad 1 que es la inicial y representa los arribos de clientes. Pueden iniciarse arribos de más de una transición, en este caso se plantean diferentes probabilidades cuya suma sea uno.

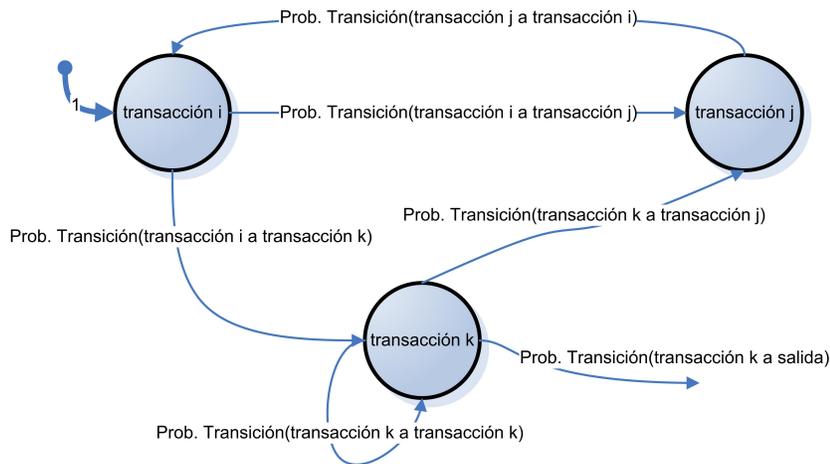


Figura 2.21: Esquema general del modelo CBMG

## 2.15. Planteo del Modelo de Recursos

El Modelo de Recursos contiene todos los componentes del sistema de computación que conforman la plataforma tecnológica y representa la infraestructura tecnológica informática y de comunicaciones que soporta el proceso de negocios.

En las vistas inferiores de la arquitectura de referencia se incluyen los recursos físicos de la plataforma informática, cuya representación es factible con la teoría de redes de cola.

Cada recurso se modela como un centro de cola o de demora, parametrizado con un tiempo de atención a cada trabajo. Si el centro está ocupado frente a un nuevo arribo, el trabajo es colocado en una cola y en este caso se tiene un centro de cola. En caso contrario el centro es de demora.

El tiempo en un centro de demora es el tiempo de servicio insumido por la atención del trabajo en el recurso. En cambio para un centro de cola a dicho tiempo se le adiciona el de espera hasta ser atendido.

La interacción entre los centros se corresponde con las diferentes sesiones de clientes. El desempeño de la red de centros se estima mediante las ecuaciones propuestas para el nivel tecnológico en la Sección 2.16.1.

Siguiendo con el ejemplo de la consulta de datos de libro a un proveedor, se definen las transacciones, recursos de hardware, sistemas informáticos y roles que identifican el responsable de la transacción en la Tabla 2.2.

Transacción	Hardware	Software	Rol del ejecutante
Visit Home Page	Computadora Personal	Servidor Web	Propietario del Sitio
Search	Computadora Personal	Servicio Web de consulta	Proveedor

Tabla 2.2: Transacciones y recursos que las ejecutan

El modelo de recursos se representa con la red de colas que se muestra en la Figura 2.22.

El servidor Web es un centro de cola que recibe arribos de requerimientos de clientes y respuestas de la consulta al servicio Web del proveedor. Los requerimientos que arriban los envía al centro de cola Internet y las respuestas las devuelve al cliente. Internet

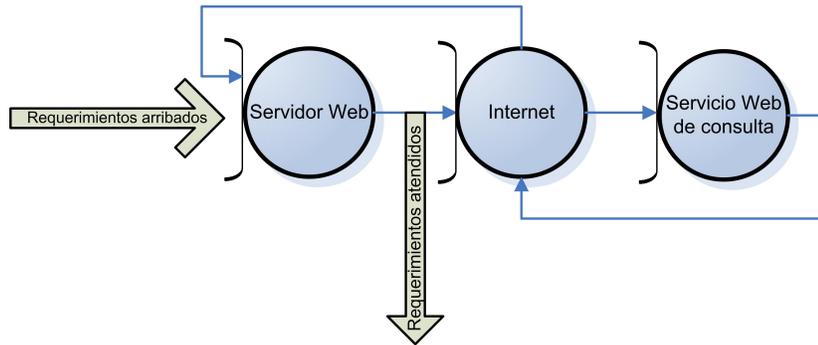


Figura 2.22: Modelo de red de colas de recursos para una consulta a una base de datos

entrega los pedidos de consulta al centro de cola de servicio Web de consulta o devuelve al servidor Web las respuestas a las mismas. Por último el servicio Web del proveedor atiende la consulta y la responde a través de Internet.

En este ejemplo se modelan tres recursos, en otros casos se podría incrementar el detalle incorporando redes locales y recursos individuales del servidor. El nivel de fragmentación es establecido por el modelador de acuerdo al interés de estudio.

En los recursos informáticos se identifican las siguientes transacciones a procesar:

- **Visit Home Page:** Gestión de la página principal del sitio.

ObtenerDetalles.Request(título):detalles

- **Search:** Gestión de la base de datos.

EnviarDetalles.Response(título):detalles

## 2.16. Esquematización de Métricas de Desempeño

En la Sección 2.6 se propuso un Framework de Métricas de Desempeño de Modelos de Negocios Electrónicos, en el cual se presentan métricas para evaluar requerimientos desde la perspectiva de tecnología informática y de comunicaciones, el producto, los clientes, los costos y los beneficios.

El detalle de las métricas se encuentra en la Tabla 2.1, de las cuales se seleccionan las necesarias para medir los requerimientos de tecnología informática y negocios y calcular los costos de inversión y los beneficios. Con los datos obtenidos se procede al análisis económico, tomando como referencia el punto de equilibrio, planificación de utilidades o análisis marginales.

Sobre la inversión realizada se espera un retorno, el cual puede requerir un análisis financiero con métricas como VNA, TIR y flujo de fondo.

Con la métrica se debe definir su indicador, esto permite no solo obtener un valor de salida sino también un instrumento de interpretación del mismo.

### 2.16.1. Enfoque Tecnológico

Desde el enfoque tecnológico se plantean las siguientes métricas básicas:

- **Velocidad de Procesamiento:** número de requerimientos de clientes completados por unidad de tiempo (requerimientos/unidad de tiempo).

En la Ecuación 2.2 se describe la relación de variables para el cálculo de la velocidad de procesamiento:

$$\text{Velocidad de Procesamiento} = \frac{\text{Cantidad de requerimientos completados}}{\text{Tiempo de Observación}} \quad (2.2)$$

- **Tiempo de Respuesta:** tiempo de residencia promedio de las sesiones de clientes en el sistema (unidad de tiempo).

Siendo  $n$  el número total de sesiones, la duración de una sesión se presenta en la Ecuación 2.3.

$$\text{Duración Sesión}_i = \text{Tiempo de Fin de Sesión}_i - \text{Tiempo de Inicio de Sesión}_i \quad (2.3)$$

La Ecuación 2.4 describe el tiempo de respuesta promedio.

$$\text{Tiempo de Respuesta Promedio} = \frac{\sum_{i=1}^n \text{Duración Sesión}_i}{n} \quad (2.4)$$

- **Utilización de los Recursos:** proporción de tiempo que el recurso está ocupado en el tiempo de observación.

Siendo:

- $X_i$  la cantidad de requerimientos por tiempo procesados en el recurso  $i$ .
- $S_i$  el tiempo de servicio demandado por un requerimiento en el recurso  $i$ .

La Ley de Utilización (Lazowska y otros, 1984) en cada observación se formula en la Ecuación 2.5.

$$U_i = X_i S_i \quad (2.5)$$

Para un número total de  $n$  observaciones, se calcula la utilización promedio como lo expresa la Ecuación 2.6.

$$\text{Utilización Promedio} = \frac{\sum_{i=1}^n U_i}{n} \quad (2.6)$$

- **Colas Promedio por recurso:** número promedio de requerimientos en cola en un recurso en espera de servicio.

La Ecuación 2.7 presenta la cantidad de requerimientos en cola de un recurso en el momento observado.

$$\text{Cola}_i = \text{Trabajos}_{\text{arribados}} - \text{Trabajos}_{\text{atendidos}} \quad (2.7)$$

En la Ecuación 2.8 se expresa la cola promedio para  $n$  observaciones en el tiempo de simulación.

$$Cola\ Promedio = \frac{\sum_{i=1}^n Cola_i}{n} \quad (2.8)$$

### 2.16.2. Enfoque de Negocios

Desde el enfoque de negocios se plantean las siguientes métricas básicas:

- **Velocidad de ingreso:** monto de ingreso promedio de ventas por unidad de tiempo [pesos/unidad de tiempo].
- **Cantidad de productos vendidos:** número de productos vendidos en un período de tiempo [productos/período].
- **Cantidad de clientes que ingresan al sitio:** número total de clientes que arriban al sistema de acuerdo a una tasa de arribo [clientes].
- **Pérdida por no confirmación de compra:** número de clientes que ingresan productos al carro de compras pero no concretan el pago.
- **Porcentajes de Transacciones por clase:** proporción de cada tipo de transacción respecto al total.
- **Sesiones más frecuentes:** caminos de transacciones seguidos por el cliente con mayor frecuencia.
- **Punto de equilibrio:** monto de ingreso necesario para igualar la inversión.

## 2.17. Construcción del Modelo DEVS

Para la especificación del modelo en el formalismo DEVS se plantean las siguientes operaciones:

1. **Construir el diagrama de estructura de las entidades del sistema.**
2. **Delinear el modelo base.**
3. **Elaborar la semántica DEVS.**

- a) Diseñar el diagrama Statechart de los modelos atómicos y acoplados.
- b) Plantear los algoritmos.

### **2.17.1. Construir el Diagrama de Estructura de las Entidades del Sistema**

Para lograr una representación modular y jerárquica se construye un diagrama de estructura que sintetiza las principales entidades en forma de árbol AVL. Se obtiene una estructura jerárquica que en forma descendente parte de lo general a lo particular, de modo que los nodos hojas se corresponden con los modelos atómicos y sus padres con los acoplados.

En la Figura 2.23 se muestra el modelo de simulación a nivel de estructuras del sistema de consulta. En la rama izquierda se observa un modelo acoplado llamado Marco Experimental, con los modelos atómicos Generador y Transductor, mientras que en la derecha se muestra el modelo acoplado Sistema Informático, formado por los modelos atómicos Servidor Web, Internet y Servicio Web de Consulta.

La estructura del sistema de consulta se construye a partir del modelo de red de colas de la Figura 2.22.

### **2.17.2. Delinear el Modelo Base**

Sobre la base del modelo de estructura se esquematizan modelos atómicos, se definen sus puertos y se articulan los acoplamientos. En la Figura 2.24 se muestra el modelo de simulación compuesto por el acoplamiento principal de los modelos Marco Experimental y

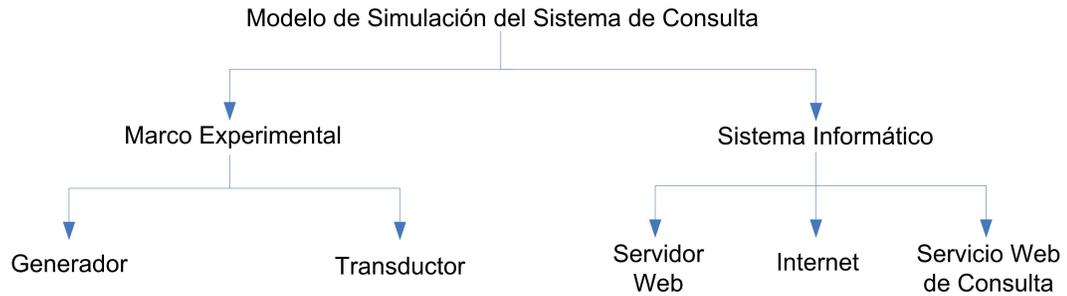


Figura 2.23: Modelo de estructura del sistema de consulta a la base de datos

Sistema Informático. En este modelo se distinguen cada uno de los puertos que son entradas y salidas de los modelos atómicos y acoplados. A través de estos puertos se establecen las conexiones para formar los acoplamientos entre modelos atómicos y acoplados.

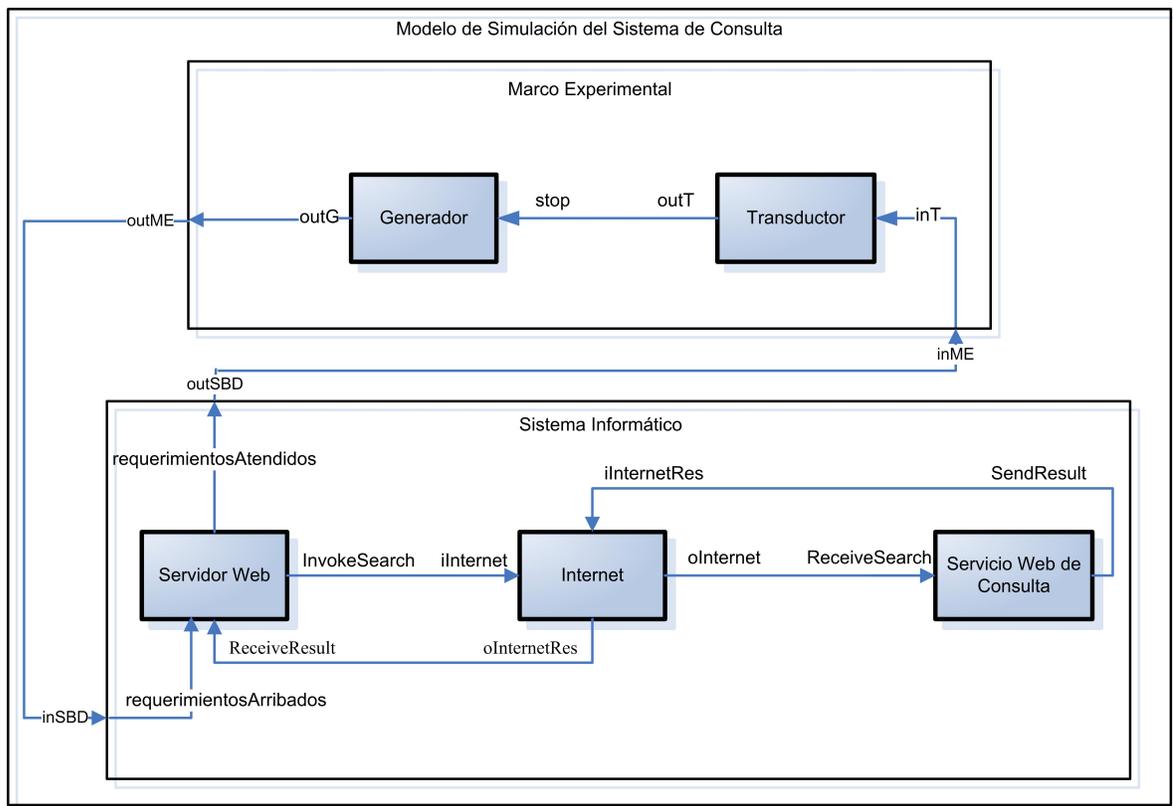


Figura 2.24: Modelo de estructura del sistema de consulta a la base de datos

El Marco Experimental envía los requerimientos generados al Sistema Informático por el

puerto de salida *outME*, los cuales son recibidos a través del puerto de entrada *inSB*. Los requerimientos atendidos son devueltos por el Sistema Informático al Marco Experimental por el puerto de salida *outBD*, los cuales son recibidos por el puerto de entrada *inME*.

El Marco Experimental es un modelo acoplado que contiene a los modelos atómicos Generador y Transductor. El Generador a través del puerto de salida *outG* produce requerimientos al Sistema Informático e informa los mismos al Transductor. Por el puerto de entrada *stop* recibe el mensaje que informa el fin de la simulación.

El Transductor tiene un puerto de entrada *inT* por el cual recibe los requerimientos atendidos y uno de salida *outT* por el cual envía el mensaje de fin de simulación.

El modelo acoplado Sistema Informático contiene los modelos atómicos Servidor Web, Internet y Servicio Web de Consulta.

El modelo atómico Servidor Web recibe requerimientos desde el modelo acoplado de consulta a través del puerto *requerimientosArribados* y le devuelve los requerimientos completados por el puerto de salida *requerimientosAtendidos*. Además, envía los requerimientos a ser atendidos al modelo atómico Internet por el puerto de salida *InvokeSearch* y recibe de este por el puerto de entrada *ReceiveResult*.

El modelo atómico Internet recibe requerimientos del Servidor Web por el puerto de entrada *iInternet* y se los devuelve por el puerto de salida *oInternetRes*. Por otro lado envía requerimientos al modelo atómico Servicio Web de Consulta por el puerto *oInternet* y recibe de este los mismos por el puerto de entrada *iInternetRes*.

Por último el modelo atómico Servicio Web de Consulta recibe los requerimientos por el puerto de entrada *ReceiveSearch* y los devuelve por el de salida *SendResult*.

Para generalizar la construcción del modelo de interacción de servicios se propone un modelo base de servicios Web, el cual se muestra en la Figura 2.25. Se muestran dos componentes atómicos, el consumidor que requiere un servicio y el proveedor que lo pone a disposición. Las operaciones básicas se indican como puertos que intercambian los mensajes request y response. Es decir, un consumidor invoca un mensaje request que es recibido por el servicio proveedor.

Se ejecuta la operación y se responde con un mensaje que espera ser recibido.

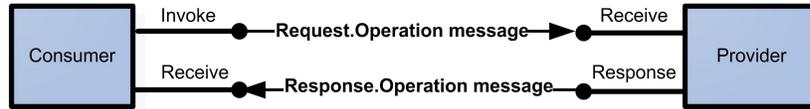


Figura 2.25: Modelo base de servicio Web

### 2.17.3. Elaborar la Semántica DEVS

Para construir la semántica DEVS se dispone del diagrama de especificación Statechart (Francis y otros, 2005). Es una herramienta orientada a gráficos, extensión de los diagramas de transición de estados, que permite especificar sistemas reactivos, con capacidades de modelado jerárquico, ortogonalidad para representación de actividades paralelas y comunicación broadcast.

En la Figura 2.26 se muestra una transición interna que consiste en una línea punteada, cuyo rótulo se interpreta como: dada una condición y cumplido un tiempo en el estado  $i$  se produce una transición al estado  $j$ , con la salida de un valor por un puerto de salida y la ejecución de una operación a tener en cuenta en el algoritmo.

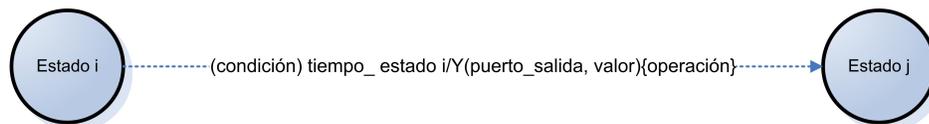


Figura 2.26: Transición interna

En la Figura 2.27 se observa una transición externa que consiste en una línea completa, cuyo rótulo se interpreta como: de acuerdo a un valor de un evento externo ingresado por un puerto de entrada se produce una transición al estado  $j$ , sin salida y se ejecuta una operación a tener en cuenta en el algoritmo.

La especificación del modelo con la herramienta Statechart y la definición formal de algoritmos se detallan en el Capítulo 3.

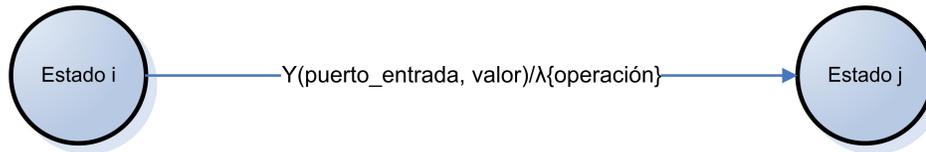


Figura 2.27: Transición externa

## 2.18. Obtención de los Parámetros para el Modelo de Simulación

Los parámetros fundamentales a tener en cuenta para la configuración del sistema son el modelo de carga de trabajo y los de los recursos computacionales de la plataforma informática.

Para modelar las cargas de trabajo se estudia el patrón de arribo de clientes al sitio de negocios y para los recursos informáticos se obtienen los tiempos de servicios.

Tanto el arribo de clientes como el funcionamiento de los sistemas informáticos están sujetos a un comportamiento no determinístico, por lo cual la naturaleza de los datos es probabilística y las variables que se generan a partir de ellos son aleatorias.

Un sistema estocástico necesita de herramientas de probabilidad para su estudio. De modo que los datos de entrada resultan de la observación de una muestra y sobre la cual se aplica un tratamiento.

El tratamiento consiste en hacer estudios estadísticos para determinar cuál es la distribución de probabilidad que los ajusta de acuerdo a un nivel de confianza acordado. La distribución teórica obtenida es utilizada para generar las variables aleatorias. Este proceso se presenta en la Figura 2.28:

Los datos se toman de archivos de Log de sistemas de monitorización de sitios de comercio electrónico.

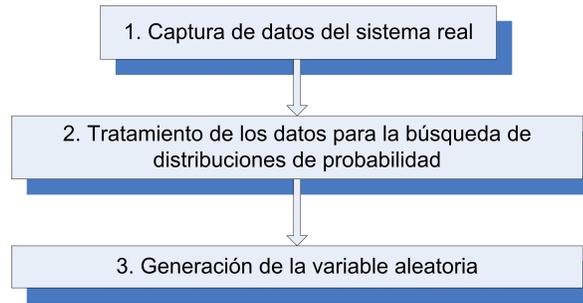


Figura 2.28: Pasos para la generación de variables aleatorias

## 2.19. Implementación en la Herramienta de Simulación

El framework DEVS puede ser implementado en DEVSJAVA, el cual es un entorno de simulación DEVS que utiliza el lenguaje Java.

DEVSJAVA es una herramienta de simulación cuya principal ventaja es la flexibilidad para la construcción de modelos personalizados de acuerdo a requerimientos específicos de cada problema.

Por sus capacidades de programación orientada a objetos, posibilita un entorno de diseño escalable y reusable del código de programa.

Respecto a la ejecución de la simulación, posee una ventana que visualiza en pasos la interacción de mensajes entre las diferentes entidades del modelo. Con la utilización de esta opción “Step” el modelador puede seguir la simulación evento a evento y rastrear su correcto funcionamiento. Además, cuenta con una opción “Run” para la simulación en un modo rápido.

## 2.20. Verificación y Validación del Modelo

La verificación del modelo consiste en determinar que el programa realiza correctamente los supuestos en la construcción del mismo. En caso contrario es necesario la corrección de

errores en la lógica del programa.

Con la verificación se demuestra la correcta ejecución del programa y para ello se aplican técnicas de debugging.

En primera instancia se ejecutan experiencias de simulación en distintos escenarios para analizar el impacto de los cambios en el modelo sobre las diferentes variables que permiten evaluar la coherencia de las salidas. En esta propuesta se trabaja con la cantidad de clientes que arriban, el tiempo de respuesta, la velocidad de procesamiento y la cantidad de clientes que compran.

Para obtener las métricas se deben generar archivos de Log por cada objeto y grabar el instante de tiempo en que se produce el evento y la descripción del mismo.

Si se encuentra incoherencia en las salidas respecto a los supuestos del modelo y las entradas, se aplican técnicas de debugging sobre el código del programa.

Otro modo de realizar la verificación es la ejecución del modelo en pasos, a medida que se acoplan nuevos componentes. La ejecución de pasos permite seguir la interacción de los mensajes y de este modo la correcta ejecución del modelo.

La validación consiste en la contrastación del modelo de simulación con el sistema fuente, para comprobar el nivel de ajuste entre ellos. Lo ideal es que el sistema fuente sea un sistema real, si esto no es posible se trabaja con prototipos de sistemas que son simplificación de los reales. Otra forma de validación es la consulta a expertos.

Por tanto se procede a comparar con datos reales si se cuenta con un sistema real, con datos teóricos si se trabaja con un prototipo y con la experiencia e intuición en caso de consulta a expertos.

Los aspectos a tener en cuenta en la validación son los valores de la variable de entrada, las distribuciones de probabilidad de las variables aleatorias, los modelos de carga de trabajo y las variables de salida.

## 2.21. Aplicación del Algoritmo de Configuraciones Eficientes

Se propone un procedimiento para la obtención de una configuración eficiente de sistemas de negocios electrónicos.

El objetivo del modelo de simulación de transacciones electrónicas es obtener métricas a nivel de tecnología informática, económicas y el punto de equilibrio. Se pretende identificar una configuración eficiente para mejorar el desempeño de sistemas de comercio electrónico. Por ello se propone un algoritmo de búsqueda que se describe en los siguientes pasos:

**Paso1:** Realizar una planificación de la capacidad.

Para tener un punto de referencia como comienzo de la ejecución de la simulación es necesario realizar una búsqueda de condiciones iniciales para el diseño experimental estadístico.

Se realiza una planificación de capacidad de los recursos informáticos con el fin de obtener tiempos de respuestas adecuados a su indicador. Para esto se proponen diferentes configuraciones de la plataforma informática, las cargas de trabajos y se planifica el proceso de simulación.

**Paso 2:** Identificar recursos críticos de la plataforma informática, desde:

**Perspectiva tecnológica:**

- Tiempo de respuesta del sistema.
- Cantidad de requerimientos en cola.
- Utilización de los recursos.

**Perspectiva de Negocios:**

- Ingresos por segundo.
- Cantidad de productos vendidos.
- Sesiones por segundos.

- Monto de inversión en T.I.

**Paso 3:** Elaboración de Indicadores.

### Indicador del Tiempo de Respuesta (IR) para comercio electrónico.

Para la elaboración del indicador del tiempo de respuesta se consideran los datos del monitor de desempeño Keynote (<http://www.keynote.com>). Del mismo se obtiene un ranking de tiempos de respuesta de sitios reconocidos de comercio electrónico tipo minorista. De las diferentes mediciones propuestas se seleccionan índices de desempeño para venta minorista de:

#### Libros:

([http://www.keynote.com/keynote\\_competitive\\_research/performance\\_indices/retail/retail\\_books\\_index.html](http://www.keynote.com/keynote_competitive_research/performance_indices/retail/retail_books_index.html)).

#### Electrónica:

([http://www.keynote.com/keynote\\_competitive\\_research/performance\\_indices/retail/retail\\_electronics\\_index.html](http://www.keynote.com/keynote_competitive_research/performance_indices/retail/retail_electronics_index.html)).

Dichas mediciones se basan en sesiones de clientes que ingresan a un sitio de comercio electrónico, buscan un producto, lo cargan al carro de compras y autorizan su pago.

De estos índices se considera el menor tiempo de respuesta de 7,95 seg. y el promedio de 10,62 seg. y se propone el Algoritmo 1.

---

### Algoritmo 1 Cálculo del Indicador del Tiempo de Respuesta

**Entrada:** Tiempo de Respuesta:  $R$ .

**Salida:** Indicador de Tiempo de Respuesta:  $IR$ .

```

1: si  $R \leq 7,95$  seg. entonces
2:    $IR = Sobresaliente$ 
3: si no
4:   si  $R > 7,95$  seg. and  $R \leq 10,62$  seg. entonces
5:      $IR = Satisfactorio$ 
6:   si no
7:     si  $R > 10,62$ seg. entonces
8:        $IR = Insatisfactorio$ 
9:     fin si
10:  fin si
11: fin si

```

---

### Indicador Económico (IE)

Para obtener el indicador económico se realiza una evaluación del punto de equilibrio. Es decir, el punto de actividad o volumen de ventas donde el total de los ingresos y el total de los costos de inversión son iguales, por tanto no hay utilidades ni pérdidas (Horngren, 1985). En la Ecuación 2.9 se presenta la expresión del punto de equilibrio.

$$\text{Costo de Inversión} = \text{Ganancia Neta} \quad (2.9)$$

La *Ganancia Neta* se expresa en la Ecuación 2.10 y consiste en obtener del monto total que ingresa por ventas, el valor neto de ganancia, discriminado los costos.

$$\text{Ganancia Neta} = \text{Ingreso por Ventas} * \text{Porcentaje de Beneficio} \quad (2.10)$$

Para obtener el *Porcentaje de Beneficio* se trabaja sobre el Estado de Resultados de la empresa, también conocido como Balance y se lo calcula como la expresión de la Ecuación 2.11.

$$\text{Porcentaje de Beneficio} = \frac{\text{Ganancia Neta del Ejercicio}}{\text{Ganancia Bruta}} \quad (2.11)$$

Utilizando la Ecuación 2.9 se propone el Algoritmo 2 para el cálculo del indicador económico.

---

**Algoritmo 2** Cálculo del Indicador Económico

**Entrada:** Cantidad de Transacciones Pagar: *CantPay*

**Entrada:** Costo de Inversión: *CI*

**Entrada:** Ganancia Neta: *GN*

**Salida:** Indicador Económico: *IE*

```

1: si  $CI > GN$  entonces
2:    $IE = Deficitario$ 
3: si no
4:   si  $CI = GN$  entonces
5:      $IE = En\ equilibrio$ 
6:   si no
7:     si  $CI \leq 1,15 * GN$  entonces
8:        $IE = Con\ utilidad$ 
9:     si no
10:      si  $CI \leq 1,20 * GN$  entonces
11:         $IE = Con\ utilidad\ adecuada$ 
12:      si no
13:         $IE = Con\ buena\ utilidad$ 
14:      fin si
15:    fin si
16:  fin si
17: fin si

```

---

El indicador no alcanza el punto de equilibrio en la Línea 1, por tanto los beneficios netos no llegan a cubrir la inversión. En la Línea 4 alcanza el punto de equilibrio, para luego obtener ganancias en diferentes niveles: un 15 % sobre el beneficio (Línea 7), 20 % sobre el beneficio (Línea 10) y un porcentaje superior a 20 % (Línea 13).

**Paso 4:** Evaluar los Indicadores del Tiempo de Respuesta y del Balance Económico.

Para las opciones de configuración simuladas se analiza el tiempo de respuesta con el IR deseado y el IE para fundamentar la inversión en tecnología.

Alcanzado los indicadores IE e IR deseados, se analiza la opción de configuración y su factibilidad de implementación. Si esta opción se acepta se cuenta con información para proponer los factores y sus niveles en el diseño experimental estadístico.

En caso contrario se deben analizar nuevas opciones de configuración, simular y volver al paso 1 para seguir evaluando otras opciones en la búsqueda de una configuración eficiente para el negocio.

## **2.22. Realización del Diseño Experimental Estadístico**

De la aplicación del algoritmo de configuraciones eficientes se obtienen los factores del diseño experimental estadístico y los valores de sus niveles. Como resultado del diseño experimental se pueden analizar los efectos significativos estadísticamente de los factores del sistema y de sus interacciones en las variables de respuesta frente a los cambios de niveles.

Los modelos de simulación de desempeño dependen de parámetros de entrada que actúan sobre el modelo y son base para la obtención de las métricas de salidas. A su vez la relación entre los parámetros de entrada y las medidas de desempeño existentes no se conocen explícitamente.

Por lo tanto, la simulación llega a ser un proceso prueba error en el cual un conjunto de parámetros de entrada es usado para predecir un conjunto de medidas de salida. Si las medidas deseadas se logran, un buen diseño de sistema ha sido encontrado, en otro caso el proceso es repetido hasta que un conjunto satisfactorio de medidas son obtenidas.

Una buena metodología de análisis de salidas de simulación no solo debe obtener los valores de las variables de respuesta sino también determinar cuáles de las entradas influyeron sobre las salidas, cómo fue esa influencia y si existe interrelación entre dichas entradas. Por ello es conveniente plantear un diseño experimental estadístico.

**“Un experimento es una prueba o serie de pruebas en las cuales se introducen cambios deliberados en las variables de entrada de un proceso o sistema, de manera que sea posible observar e identificar las causas de los cambios en la respuesta de salida” (Montgómery, 2001).**

El objetivo del experimento es estudiar el efecto de los factores, lo que consiste en determinar cuáles variables tienen mayor influencia en la respuesta y cómo es esa influencia. El diseño de experimentos estudia el sistema en su conjunto, se exploran las causas llamadas factores que provocan el problema desde un enfoque sistémico y estadístico, es decir se busca tener una relación causa-efecto (Montgómery, 2001; Bartés y Fernández, 2000).

La metodología de trabajo parte de una elección de factores y sus niveles, como parámetros de entrada y sus valores y la selección de las variables de respuesta o métricas de salidas. De acuerdo a la cantidad de factores y niveles, se selecciona el diseño experimental a aplicar. Se realiza el experimento, se obtiene el análisis de varianza ANOVA y se hacen las conclusiones y recomendaciones pertinentes.

El Factorial Completo es uno de los modelos de diseño. En el mismo se consideran todas las posibles combinaciones de los niveles de los factores en cada réplica del experimento y se estudia el efecto de dos o más factores en la respuesta. En la Ecuación 2.12 se presenta el Modelo Estadístico Lineal de la variable de respuesta para tres factores.

$$y_{ijkl} = \mu + \tau_i + \beta_j + \gamma_k + \tau_i\beta_j + \tau_i\gamma_k + \beta_j\gamma_k + \tau_i\beta_j\gamma_k + \epsilon_{ijkl} \begin{cases} i = 1, 2, \dots, a \\ j = 1, 2, \dots, b \\ k = 1, 2, \dots, c \\ l = 1, 2, \dots, n \end{cases} \quad (2.12)$$

Donde:

$y_{ijkl}$  es la  $(ijk)$ -ésima observación para el tratamiento  $l$ .

$\mu$  es la media global (común a todos los tratamientos).

$\tau_i$  es el efecto del  $i$ -ésimo nivel del factor A.

$\beta_j$  es el efecto el  $j$ -ésimo nivel del factor B.

$\gamma_k$  es el efecto del  $k$ -ésimo nivel del factor C.

$\tau_i\beta_j$  es el efecto de la interacción de los factores A y B.

$\tau_i\gamma_k$  es el efecto entre la interacción de los factores A y C.

$\beta_j\gamma_k$  es el efecto entre la interacción de los factores B y C.

$\tau_i\beta_j\gamma_k$  es el efecto entre la interacción de los factores A, B y C.

$\epsilon_{ijkl}$  componente aleatoria del error.

$a$  niveles del factor A.

$b$  niveles del factor B.

$c$  niveles del factor C.

$N$  observaciones.

$n$  número de réplicas para un nivel.

Sobre el modelo de la Ecuación 2.12 se realiza un test de hipótesis con el fin de analizar los efectos de los tratamientos sobre la variable de respuesta y determinar su grado de significación.

El test de hipótesis consiste en probar la igualdad de las medias de los tratamientos, de modo que para  $\mu_1$  la media del tratamiento 1,  $\mu_2$  la media del tratamiento 2 y  $\mu$  la media de los tratamientos:

La hipótesis nula  $H_0 : \mu_1 = \mu_2 = \dots = \mu$

La hipótesis alternativa  $H_1 : \mu_i \neq \mu_j$  al menos para un par  $(i, j)$

Si  $H_0$  es verdadera todos los tratamientos tienen una media común  $\mu$ , en caso contrario al menos uno de los tratamientos va a tener una media diferente y por tanto es

verdadera la hipótesis alternativa.

Si todos los tratamientos tienen la media común los efectos no varían en los cambios de niveles de un factor o en la interacción de factores y por tanto no son significativos sobre la variable de respuesta. En caso de su rechazo, por lo menos una de las medias de los niveles del factor es diferente de  $\mu$  y por tanto los efectos son significativos para la variable de respuesta.

El método para probar la igualdad de las medias es el análisis de varianza llamado ANOVA. El criterio para aceptar o rechazar la hipótesis nula se basa en la evaluación del valor  $P$  o de la variable  $F$  de Fisher, rechazándose la hipótesis nula si el valor  $P$  obtenido es menor que el nivel de significación  $\alpha$  elegido o si la variable  $F$  para cada factor e interacciones es mayor que el correspondiente  $F$  de tabla.

La descomposición de la variabilidad en las observaciones por medio del análisis de varianza es una relación puramente algebraica. Sin embargo se requiere que se satisfagan ciertas suposiciones con el objeto de probar formalmente la igualdad en el nivel medio de los tratamientos, tales como:

- Los errores del modelo son variables aleatorias independientes, normalmente distribuidos, con media cero y varianza  $\sigma^2$ .
- $\sigma^2$  es constante para todos los niveles del factor.

Para la ejecución del experimento se plantean los siguientes pasos:

1. Planteo del problema y objetivo del experimento.
2. Estimación del tamaño de la muestra.
3. Selección de los factores y sus niveles: se deben identificar todas las posibles fuentes de variación. Una fuente de variación es cualquier factor que pueda generar

variabilidad en la respuesta. Es recomendable hacer una lista de todas las posibles fuentes de variación del problema, distinguiendo aquellas que, a priori, generarán una mayor variabilidad.

4. Elección de las variables de respuesta.
5. Propuesta del diseño experimental estadístico.
6. Comprobación de la idoneidad del modelo

a) Suposición de Normalidad.

Un procedimiento posible consiste en construir el gráfico de la probabilidad normal de los residuos, que representa la distribución acumulada de los residuos sobre la probabilidad normal, es decir sobre un gráfico donde la escala de ordenadas es tal que la distribución normal sea una recta. Si los errores  $\epsilon_{ijkl}$  están  $N(0, \sigma^2)$ , los residuos estandarizados deben ser aproximadamente normales con media cero y varianza uno. Por ello el 68 % de los residuos estandarizados deben encontrarse entre los límites  $\pm 1$ , el 95 % entre  $\pm 2$  y todos deben estar entre  $\pm 3$ .

b) Gráfico de residuos versus el orden de los datos observados en el tiempo.

c) Gráfico de residuos versus el valor ajustado (valor promedio del nivel).

d) Prueba estadística para mostrar la igualdad de la varianza. A través de la prueba de Bartlett se puede mostrar la igualdad de la varianza como otra técnica.

7. Análisis de resultados y planteo de nuevos experimentos.

De la evaluación del costo de simulación se pueden plantear otros diseños experimentales que minimicen el mismo respecto al Factorial Completo.

## **2.23. Análisis de Resultados**

Con los datos de simulación se aplica el Algoritmo de Configuraciones Eficientes y luego el Diseño Experimental Estadístico, de lo que se obtiene información de las métricas mediante gráficos de resultados.

El análisis de resultados es un trabajo cíclico que puede llevar a la búsqueda de nuevos escenarios y por tanto a nuevos diseños experimentales.

Una vez encontrada la configuración informática adecuada a los indicadores de desempeño tecnológico y el retorno de la inversión se obtienen las conclusiones.

La aplicación de las etapas de esta metodología no es estrictamente secuencial, en cada una de ellas se pueden encontrar problemas al hacer redefiniciones que obliguen a retroceder.

## **2.24. Conclusiones**

El capítulo comienza con la conceptualización del modelo de negocios, del modelo de proceso de negocios y de los negocios electrónicos. Del modelo de negocios se destaca la propuesta de una estrategia, objetivos para el logro de la estrategia, asociaciones y transacciones que los llevan adelante, con la posibilidad de analizar oportunidades de negocios y planificando fuentes de ingreso de dinero.

Los modelos de negocios son implementados en un proceso de negocios, que por utilizar recursos informáticos y de comunicaciones como estrategias se transforman en electrónicos. La estrategia de negocios puede tener como fin el marketing, la promoción

de productos, los catálogos hasta las operaciones totalmente virtuales de comercio.

La implementación de procesos de negocios electrónicos tiene asociada inversión en tecnología y por tanto es importante conocer las posibilidades de retorno de la inversión. Por ello se analizan las posibilidades de medición de desempeño desde una doble perspectiva, métricas tecnológicas por un lado y de negocios por otro.

Para la obtención de métricas se muestran las técnicas y las herramientas de medición con las aplicaciones informáticas, la utilización de los sitios Web y la planificación de su capacidad.

Como resultado del trabajo sobre las diferentes herramientas se propone un Framework de Métricas Integradas de Desempeño. Se detallan un conjunto de métricas para diferentes grupos de requerimientos, de modo que a partir del mismo se determinen costos y beneficios del proceso de modelo de negocios. A través de la ecuación del punto de equilibrio se encuentra un balance entre los costos de inversión y las ganancias netas. Se obtiene el período de retorno de la inversión y si es necesario se aplican medidas financieras para actualizar el valor del dinero.

De las diferentes técnicas se utiliza la simulación de evento discreto para la construcción de modelos de evaluación de desempeño de procesos de comercio electrónico.

Los procesos de comercio electrónico requieren de la interacción con socios de negocios, aplicaciones distribuidas y de bajo acoplamiento, los cuales se diseñan en una arquitectura orientada a servicios. Por tanto se describen los componentes básicos de una arquitectura SOA como base para el desarrollo de la metodología de modelado y simulación propuesta.

Como primera aproximación al análisis de desempeño de modelos de procesos de comercio electrónico con simulación, se trabaja con un soporte de Workflow para un modelo tipo B2C que integra los requerimientos de negocios con los servicios que los implementan. A partir de la simulación del proceso en la herramienta TIBCO se analiza

su desempeño sobre la base de métricas del tiempo promedio de respuesta de los casos en el sistema y las utilizaciones de los servicios. Además, se obtiene la cantidad de clientes que compran, lo cual permite el cálculo de los ingresos.

La simulación de procesos permite evaluar la lógica de negocios sin interrumpir la ejecución de los procesos corrientes y pronosticar comportamientos frente a cambios, lo cual la hace una herramienta poderosa que puede predecir futuros comportamientos y evitar pérdidas económicas.

Se elabora un framework de modelización para simulación orientado a servicios y se propone la metodología de modelización para simulación de transacciones de comercio electrónico.

En resumen, las etapas consisten en: (i) determinar los objetivos y la estrategia en base a los recursos informáticos y de comunicaciones con el fin de elaborar el modelo de negocio, (ii) diseñar el proceso de negocios que representa las transacciones internas a través de la orquestación de transacciones electrónicas propias y las interacciones con asociados por medio de la coreografía con redes de socios (iii) configurar la plataforma informática, (iv) proponer las métricas integradas, (v) elaborar el modelo DEVS e implementarlo en DEVSJAVA, (vi) aplicar el algoritmo de configuraciones eficientes y el diseño experimental estadístico y (vii) analizar los resultados de simulación y proponer la solución.

El algoritmo de configuraciones eficientes ayuda a encontrar una plataforma informática que a su vez demuestre posibilidad de retorno de inversión. Respecto a los indicadores se destaca que los tecnológicos se deben actualizar continuamente por los avances propios y que la selección de los económicos depende de la magnitud de la empresa y del margen de ingreso que produce el tipo de rubro de negocios.

Por otro lado, el período del retorno de la inversión y los porcentajes esperados de beneficios se determinan en función de las decisiones de cada empresa en particular.

# CAPÍTULO 3

## Especificación de Sistemas de Comercio Electrónico con Técnicas Conducidas por Eventos Discretos

*En este capítulo se presentan diferentes teorías para la modelización de sistemas de eventos discretos y se profundiza la descripción de las utilizadas en la metodología propuesta en el Capítulo 2. Entre ellas se describen el diagrama CBMG, Redes de Cola, Redes de Petri, Statechart y Formalismo DEVS. Para cada teoría se analiza su utilización en el diseño de sistemas informáticos de procesos de comercio electrónico. Se desarrolla el Formalismo DEVS Tradicional y se presentan sus extensiones al caso paralelo y estocástico. Se aborda la descripción formal del Framework DEVS y se describen las entidades básicas para la organización del modelo de proceso de comercio electrónico en los niveles transaccional y tecnológico. Para cada entidad se construye un diagrama Statechart, el cual es base para la elaboración de la lógica formal. Con lo propuesto en este capítulo se cuenta con una formalización de la semántica DEVS para la construcción de modelos de proceso de comercio electrónico.*

### 3.1. Modelos de Sistemas de Eventos Discretos

En la Sección 1.3 de la Introducción se presenta el concepto de sistemas y se plantea el estudio de sistemas informáticos para analizar su desempeño desde la perspectiva integrada de recursos y requerimientos de usuarios. La estrategia de análisis se fundamenta en la simulación y por tanto se requiere de modelos que representen el objeto de estudio.

Dichos modelos se categorizaron como simbólicos, dinámicos y probabilísticos. Es decir, el modelo se construye con capacidades para mostrar cambios de estados en diferentes pasos del tiempo. Por las características del problema de interés, los pasos del tiempo son aleatorios y por tanto las variables del sistema pueden tomar diferentes valores. De este modo se concibe el sistema como discreto.

Respecto a los cambios de estados, se pueden considerar dos situaciones: (i) ocurrencias de ingresos externos en una instancia de tiempo llamados eventos y (ii) finalización de una acción que estaba en ejecución (actividad) (Khoshnevis, 1994).

Por tanto se va a modelar un proceso de comercio electrónico desde un enfoque sistémico, como conjunto de componentes interconectados que tienen como finalidad común realizar una función de negocios.

En cualquier punto del tiempo se puede caracterizar el estado del sistema y analizar el valor de variables de interés de acuerdo a parámetros establecidos. La dinámica del sistema se construye en base a la capacidad de cambiar de estado como respuesta a eventos externos o actividades que finalizan su ejecución.

## 3.2. Diagramas de Transición de Estados

Los procesos de comercio electrónico pueden ser modelados por diagramas de transición de estados y para comprender su fundamento es conveniente partir del concepto de autómata.

Un autómata finito  $A$  se define por la 5-tupla:

$$A = \langle S, \Sigma, S_0, \delta, F \rangle$$

Donde:

- $S$  es el conjunto finito de estados.
- $\Sigma$  es un alfabeto finito.
- $s_0 \in S$  es el estado inicial.
- $\delta : S \times \Sigma \rightarrow S$  es una función de transición de estados.
- $F \subseteq S$  es el conjunto de estados finales.

Un autómata finito recibe a través de su estado inicial una cadena de entrada, de acuerdo a un alfabeto establecido. En función de cada símbolo ingresado y de una función de transición se produce un cambio de estado. Si al finalizar el proceso de la cadena de entrada, el autómata se detiene en un estado final de aceptación, se dice que la cadena pertenece al alfabeto definido. El estado inicial  $s_0$  es único, mientras que los finales pueden ser más de uno.

Un autómata finito es determinístico si para cada estado  $q \in S$  y cualquier símbolo  $a \in \Sigma$ , existe una y sólo una transición posible  $\delta(q, a)$ . En caso contrario es no determinístico.

Un autómata finito puede ser representado como un digrama de transición de estados finitos en un grafo. Este grafo contiene nodos que representan sus estados y aristas dirigidas que definen la transición entre estados.

En la Figura 3.1 se presenta un diagrama no determinístico con tres estados, de los cuales  $S_1$  es el estado inicial y  $S_3$  es el final. Sobre cada arista se identifican las transiciones  $t_1$ ,  $t_2$ ,  $t_3$  y  $t_4$ .

Se puede definir la siguiente función de transición de estados:

$$\delta = \{(S_1, t_1, S_2), (S_2, t_2, S_2), (S_1, t_3, S_3), (S_2, t_4, S_3)\}.$$

En el estado  $S_2$  se presentan dos transiciones  $t_2$  y  $t_4$ .

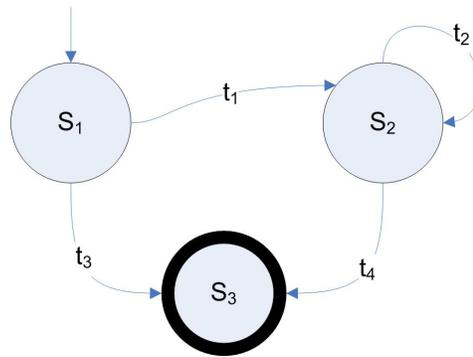


Figura 3.1: Grafo de Transición de Estados

En procesos de comercio electrónico estos diagramas caracterizan el comportamiento de clientes en el sitio de negocios, el cual se identifica por sesiones.

Las sesiones de clientes consisten en los diferentes tipos de secuencias de transacciones seguidas por los clientes durante su navegación en el sitio Web de negocios.

El Customer Behavior Model Graph (CBMG) (Menascé y otros, 2000) es una aplicación del autómata de estados finitos no determinístico y consiste en un grafo de transición de estados finitos que describe las posibles sesiones de clientes en el sitio.

En el mismo, cada estado es una transacción electrónica y cada arista es una transición a la próxima de acuerdo a una probabilidad. Por tanto su función de transición

es no determinística y está sujeta a una probabilidad de ocurrencia. En un estado, el conjunto de transiciones tiene probabilidades cuya suma es uno.

En la Figura 3.2 se muestra un grafo con cuatro estados y sus probabilidades de transición. Visitar página principal es el estado inicial a partir del cual se puede seguir a Buscar con una probabilidad  $P_1$ , Navegar con una probabilidad  $P_2$  y Salir con una probabilidad  $P_3$ . La suma de las probabilidades  $P_1$ ,  $P_2$  y  $P_3$  deben ser uno.

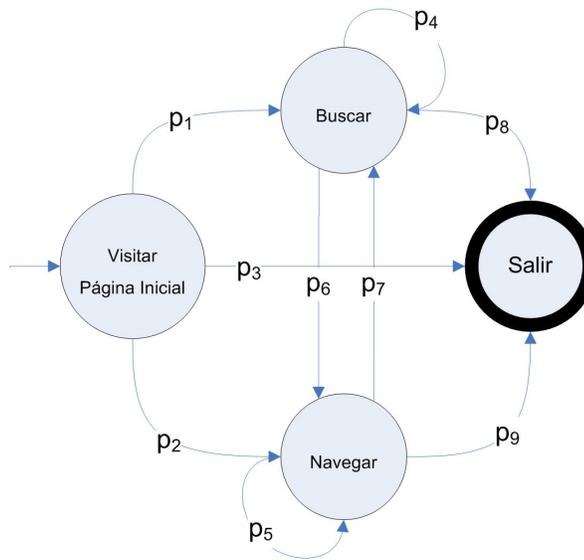


Figura 3.2: Customer Behavior Model Graph General

Un autómata finito *CBMG* se define por la 5-tupla:

$$CBMG = \langle S, \Sigma, S_0, \delta, F \rangle$$

Donde:

- $S = \{ \text{Visitar Página Inicial (VPI)}, \text{Buscar}, \text{Navegar}, \text{Salir} \}$
- $\Sigma = \{ \text{cliente}_1, \text{cliente}_2, \dots, \text{cliente}_n \}$ .
- $VPI \in S_0$

- $\delta = \{(VPI, P_1, Buscar), (VPI, P_2, Navegar), (VPI, P_3, Salir),$   
 $(Buscar, P_4, Buscar), (Buscar, P_6, Navegar), (Buscar, P_8, Salir),$   
 $(Navegar, P_5, Navegar), (Navegar, P_7, Buscar), (Navegar, P_9, Salir)\}.$   
 con  $P_1 + P_2 + P_3 = 1, P_4 + P_6 + P_8 = 1, P_5 + P_7 + P_9 = 1$
- $Salir \subseteq Q$

El CBMG se utiliza para modelar un patrón navegacional de clientes, elaborar modelos de cargas de trabajos a sitios de negocios electrónicos, responder a preguntas del tipo “¿qué pasaría si?” frente a cambios tales como parámetros del modelo, el layout del sitio o el rediseño de sus contenidos (Almeida y Menasce, 2002).

En la Figura 3.3 se presenta un caso ejemplo propuesto por Almeida (2002). Los estados se corresponden con las transacciones Visitar Página Inicial, Navegar, Buscar, Seleccionar, Agregar al Carro de Compras y Pagar. El estado Salir no se muestra explícitamente con un nodo, porque puede suceder en cualquiera de los otros estados, es decir, el cliente puede abandonar el sitio con una sesión que no finaliza en una compra.

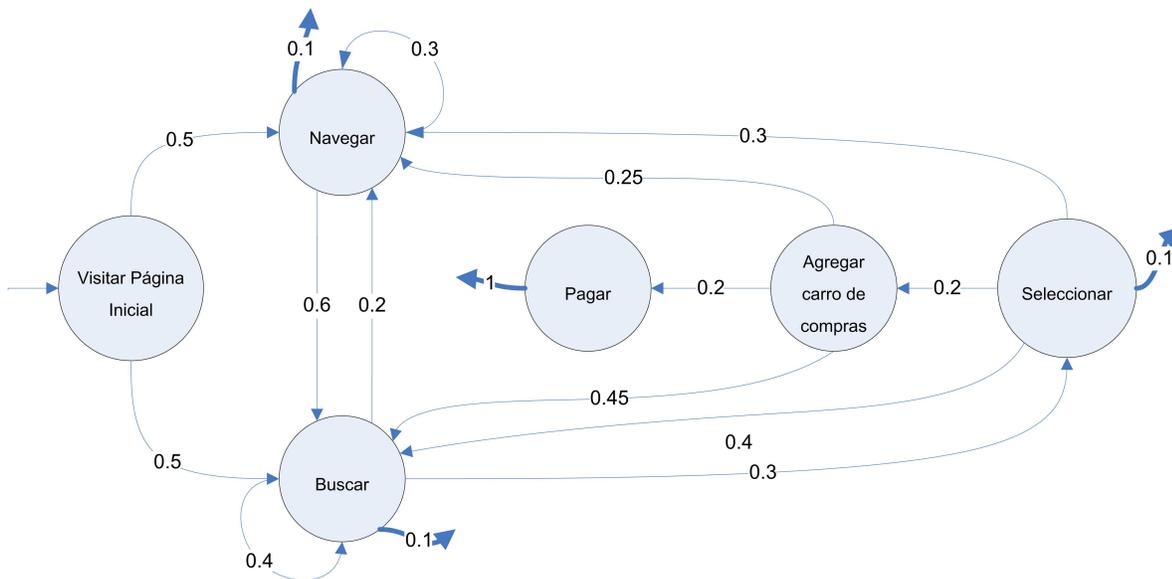


Figura 3.3: Ejemplo de Customer Behavior Model Graph

A partir del CBMG se calculan las visitas a los diferentes estados o transacciones, definiendo el modelo de carga de trabajo en el sitio de negocios.

Siendo  $V_j$  el número promedio de veces que el estado  $j$  del CBMG es accedido por cada visita al sitio de comercio electrónico, entonces:

$$V_{Agrega} = V_{Selecciona} \times 0,2 \quad (3.1)$$

En la Ecuación 3.1 se observa que el número promedio de visitas al estado Agregar es igual al promedio de visitas al estado Seleccionar multiplicado por la probabilidad 0.2 de que un cliente que está en Seleccionar pase a Agregar.

$$V_{Navega} = V_{Busca} \times 0,2 + V_{Selecciona} \times 0,3 + V_{Agrega} \times 0,25 + V_{Navega} \times 0,3 + V_{VPI} \times 0,5 \quad (3.2)$$

El número promedio de visitas al estado Navegar se detalla en la Ecuación 3.2 y se obtiene como la sumatoria del producto de las visitas promedios de transacciones que tienen una transición a Navegar multiplicada por su probabilidad de transición.

Por tanto, el número promedio de visitas a un estado  $j$  de un CBMG es igual a la sumatoria del número de visitas de todos los estados del CBMG multiplicados por la probabilidad de transición de cada estado al  $j$ . Para cualquier estado  $j$  ( $j = 2, \dots, n-1$ ) de un CBMG la Ecuación 3.3 especifica el cálculo de  $V_j$ . No se incluye el estado  $n$  ya que se corresponde al estado Salir.

$$V_j = \sum_{k=1}^{n-1} V_k \times p_{k,j} \quad (3.3)$$

Siendo  $p_{k,j}$  la probabilidad de que un cliente haga una transición del estado  $k$  al  $j$ .

El estado Visitar Página Inicial es el ingreso y su probabilidad es 1 en el CBMG. Se puede encontrar el número promedio de visitas a cada transacción resolviendo el

sistema de ecuaciones lineales de 3.4 y 3.5

$$V_1 = 1 \quad (3.4)$$

$$V_j = \sum_{k=1}^{n-1} V_k \times p_{k,j} \quad j = 2, \dots, n-1 \quad (3.5)$$

Por definición el  $V_n = 1$ , ya que el estado de salida es visitado una vez por sesión.

Además de obtener el número promedio de visitas a cada estado, se puede calcular la longitud de sesión promedio (Ecuación 3.6).

$$\text{Longitud Sesión Promedio} = \sum_{j=2}^{n-1} V_j \quad (3.6)$$

De la solución analítica del CBMG se obtiene un esquema de carga a los recursos informáticos de cada una de las transacciones de un sitio de negocios. Además, se pueden identificar las sesiones de clientes que demandan mayor cantidad de visitas a través de la longitud de sesión promedio. Con estos datos se determina los recursos que tienen mayor acceso por parte de los clientes.

### 3.3. Teoría de Redes de Colas

En los sistemas informáticos cada uno de los recursos que los componen pueden ser representados como colas, de modo que un requerimiento que arriba al mismo, es atendido si este está libre o puesto en cola en caso contrario. Además, el tiempo de respuesta de un requerimiento es la suma del tiempo que estuvo en cola y del tiempo de atención en el servicio.

Por tanto los sistemas informáticos pueden ser representados por centros de colas. La teoría de colas se basa en el modelo general de colas formado por un centro de

servicio y una cola de espera de aquellos clientes que no pueden ser atendidos.

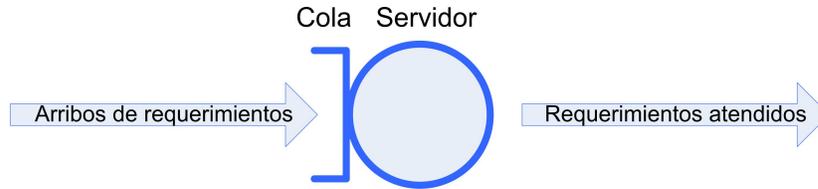


Figura 3.4: Centro de Servicio Simple

En la Figura 3.4 se muestra un modelo de cola, el cual contiene dos parámetros: la tasa de arribo de los clientes que representan la intensidad de la carga de trabajo en el sistema y los requerimiento de servicio promedio de un cliente en el recurso o demanda de servicio. Con ecuaciones de las Leyes Fundamentales (Lazowska y otros, 1984) se calculan métricas de desempeño tales como utilización del recurso, tiempo promedio del requerimiento en el sistema o 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 (Puigjaner y otros, 1992).

La teoría de colas considera 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 colas. Un centro de servicio simple se mostró en la Figura 3.4, el cual se constituye por una cola y un servidor, de modo que un requerimiento que arriba al centro de servicio, es asignado al servidor si está desocupado, de lo contrario espera en cola hasta que se desocupe. Una vez que el requerimiento es atendido se produce su salida.

Para parametrizar un centro se especifica una intensidad de carga y el tiempo que demanda en promedio un servicio en el mismo. Por las características de algunos recursos y su intensidad de arribo no requieren de una cola, para ese caso se plantea un centro de demanda como se muestra en la Figura 3.5.

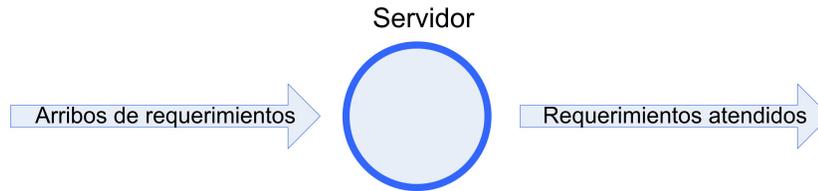


Figura 3.5: Centro de Demanda

Un sistema informático se forma por varios recursos principalmente el procesador, los discos, las redes de conexión entre otros, por ello para construir un modelo se requiere de un conjunto de centros de servicio o demanda. Es decir, las salidas de un centro de servicio o de demanda pueden ser entrada de otro, así como las salidas de más de un centro de servicio o demanda pueden ser combinadas en una entrada de otro, constituyendo así una red de colas.

La teoría de redes de colas es el fundamento para la construcción de modelos como una red de colas y la evaluación de su comportamiento mediante técnicas analíticas de solución (Lazowska y otros, 1984).

En la Figura 3.6 se observan dos recursos, un servidor y un disco, modelado como dos centros de servicios. Los requerimientos arriban al sistema y son atendidos por el servidor, si requieren acceso al disco son enviados al mismo quien luego de atenderlo lo devuelve al servidor. Si el servidor finalizó la atención del requerimiento es devuelto como salida.

Para parametrizar el modelo se requieren la tasa de arribo de requerimientos al sistema, el tiempo de demanda de servicio promedio de un requerimiento en el servidor y en el disco. Se considera que el modelo es abierto porque no se identifican los clientes o los mismos hacen visitas esporádicas al sistema y solo se pueden parametrizar con una tasa de arribos.

Respecto a la carga de trabajo, los modelos pueden clasificarse como clase simple,

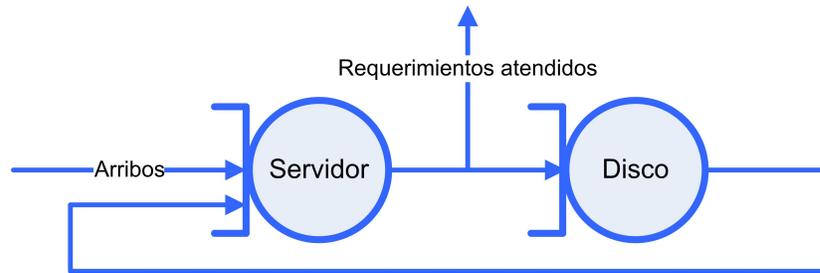


Figura 3.6: Modelo de red de colas abierta

con un único tipo de trabajo y clases múltiples con diferentes tipos de trabajos.

En la Figura 3.7 se propone un cambio respecto a los arribos. En este caso se cuenta con un número conocido de usuarios indicados como centros de demanda. El modelo se dice que es cerrado y cada usuario tiene asociado un tiempo de pensado.

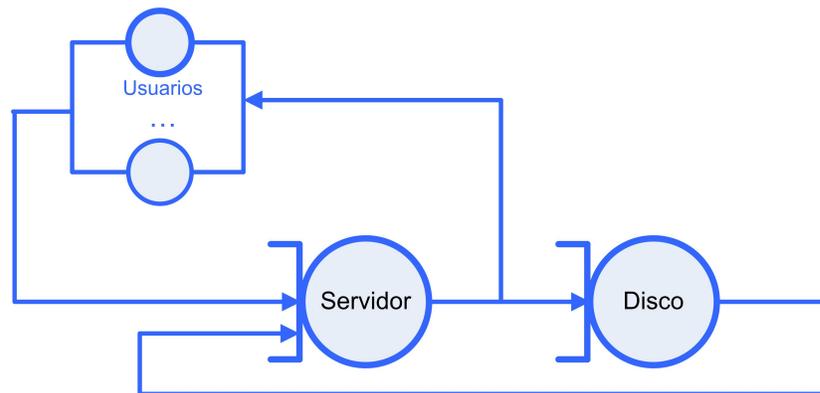


Figura 3.7: Modelo de red de colas cerrada

La ventaja de las redes de colas es su capacidad para describir los procesos de colas de un sistema, con claridad en el diseño y facilidad para representar el flujo de un cliente.

Como ejemplo de aplicación se muestra en la Figura 3.8 un modelo de red de colas abierta de un servidor de archivos. Un requerimiento que arriba es recibido por el procesador de entradas, el cual es colocado en cola si está ocupado, de lo contrario es atendido. Una vez finalizada la atención en el procesador de entradas, el requerimiento

es atendido por el router y enviado al servidor, el cual puede ser puesto en cola o atendido. El servidor requiere de una visita al disco para luego enviar el requerimiento al procesador de salidas.

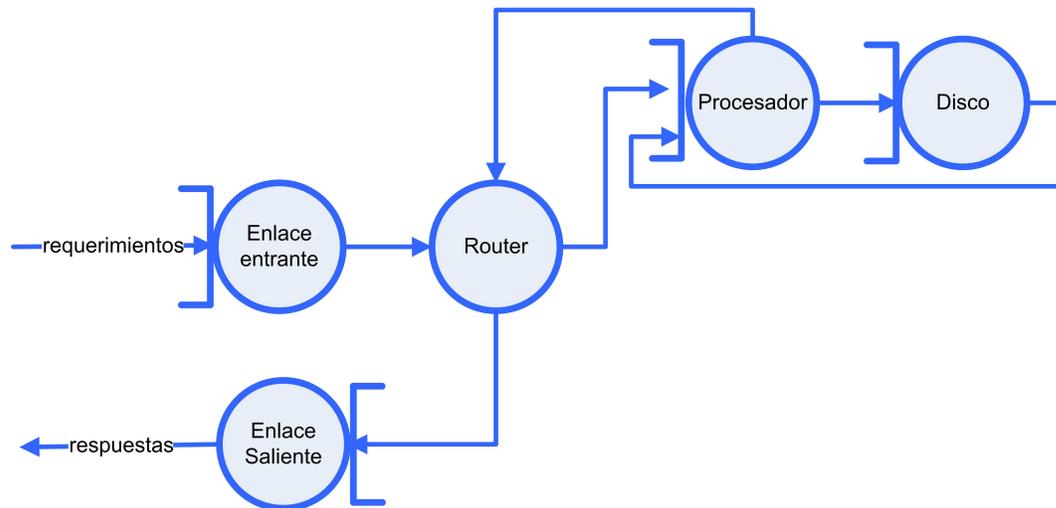


Figura 3.8: Red de colas de un servidor Web

Por tanto, si las aplicaciones pueden ser modeladas como colas y servidores interconectados, la teoría de redes de colas es adecuada.

Las redes de colas tienen la característica de ser simples de representar y muestran claridad respecto al seguimiento de la secuencia de los requerimientos en los diferentes recursos. En cambio si la representación del sistema necesita mayor detalle, en el sentido de identificar eventos que muestren la dinámica del sistema, el tratamiento tiene una importante relación con la ocurrencia de eventos, el modelo es de naturaleza reactiva y son ellos los responsables por cambios entre estados.

En estos casos se agrega complejidad y es recomendable usar técnicas con características adicionales como: eventos y sus efectos, jerarquía y refinamiento de estados, paralelismo y concurrencia entre estados. Por tanto, herramientas tales como Redes de Petri y diagramas Statecharts, son apropiadas.

### 3.4. Redes de Petri

Las redes de Petri son un tipo de grafos dirigidos que se construyen en función de cuatro elementos: places, transiciones, arcos y tokens. Su estructura algebraica es:

$$P = \{P, T, E, S\}$$

Donde:

- $P = p_1, p_2, \dots, p_n$  es el conjunto de places.
- $T = t_1, t_2, \dots, t_m$  es el conjunto de transiciones.
- $E = PxT \rightarrow N$  es una función de entrada que especifica el place de entrada de una transición, con  $N = 1, 2, \dots, n$ .
- $S = PxT \rightarrow N$  es una función de salida que especifica el place de salida de una transición, con  $N = 1, 2, \dots, n$ .

La red de Petri como grafo se esquematiza con dos tipo de nodos, places y transiciones. Los places se grafican como círculos y las transiciones como barras verticales.

Un place representa un estado que indica donde el sistema está en un momento dado. Un arco de un place  $p$  a una transición  $t$  define una entrada a la transición, mientras que un arco de una transición  $t$  a un place  $p$  define la salida de la transición.

Una transición indica un cambio de estado basado en la ocurrencia de un evento. La ocurrencia de un evento se representa por medio del movimiento de un token desde un place de entrada a un place de salida de una transición dada.

En la Figura 3.8 se propuso un modelo de servidor representado como redes de colas, el mismo se plantea como una red de Petri en la Figura 3.9.

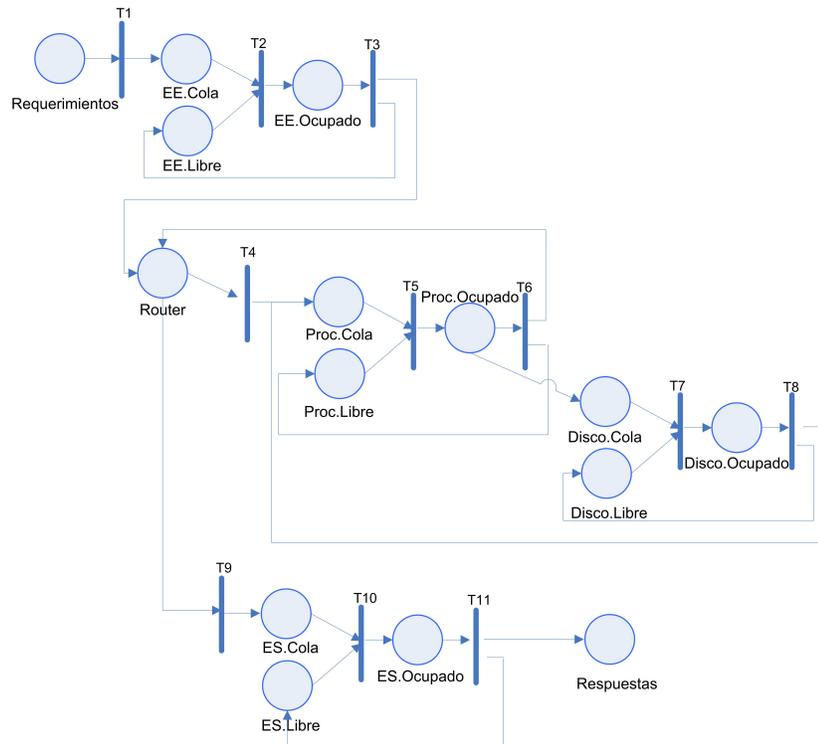


Figura 3.9: Red de Petri de un servidor Web

De la comparación de redes de cola y redes de Petri se observa que para cada uno de los recursos, tales como enlace entrante y saliente, router, servidor y disco definidos en la primera, sus estados son detallados en la segunda.

Un requerimiento que arriba produce la transición  $T_1$  activando la cola del enlace entrante *EE.Cola*. Si el enlace entrante está libre *EE.Libre* y hay un requerimiento en cola para el mismo se habilita la transición  $T_2$  y pasa al enlace entrante ocupado *EE.Ocupado*. Esto dispara la transición  $T_3$  con el *Router*, para luego ingresar al procesador. El razonamiento es idéntico para el resto del modelo.

Para una representación más compacta y de este modo describir sistemas complejos, se asocia al token valores de datos llamados tokens coloreados, como se observa en las redes de Petri jerárquicas y coloreadas.

Si bien la especificación con redes de Petri son adecuadas para este tipo de modelado, se tornan complejas cuando el modelo crece en el número de recursos y en clases de trabajo múltiples.

### 3.5. Statecharts

Es un técnica orientada a gráficos aplicada para especificar sistemas reactivos. Se los puede considerar una extensión de los diagramas de transición de estados, pero agregando funcionalidades de jerarquía, ortogonalidad e interdependencia.

A través de la jerarquía se es capaz de abordar complejidad, con la ortogonalidad se articulan actividades paralelas y con la interdependencia se hace posible una comunicación broadcast. Los elementos básicos para representar un sistema son: estado, evento, condición, acción y transición.

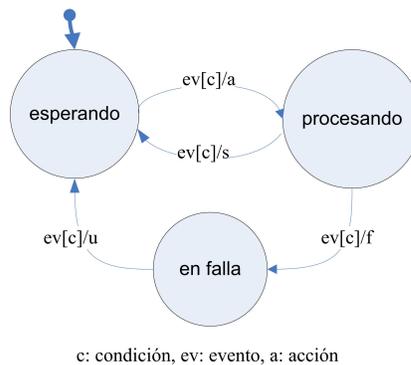


Figura 3.10: Statechart de un servidor

En el ejemplo de la Figura 3.10 se muestran los elementos básicos de representación de un servidor, los cuales se describen a continuación:

- **Estados:** describen un aspecto de un sistema dado.

1. El estado inicial es “esperando”.
  2. “Procesando”el trabajo arribado.
  3. El servidor puede fallar mientras está procesando, el nuevo estado es “en falla”.
- **Eventos:** son ocurrencias de acciones de entrada que individualmente provocan un cambio en el estado del sistema.
    1. Arribo de un evento: el sistema permanece en el estado “esperando” hasta el arribo de un requerimiento, ante el cual cambia al estado “Procesando”.
    2. Fin del servicio: finaliza el procesamiento y el sistema retorna a “esperando”.
    3. Falla: el sistema pasa al estado “en falla”.
    4. Recuperación de una falla: el sistema pasa de “en falla” a “esperando”.
  - **Acción:** es un cambio disparado por la ocurrencia de un evento.
  - **Condición:** de manera opcional la acción puede estar sujeta, no sólo a un evento sino además al cumplimiento de una condición.
  - **Transición:** es la representación física que indica un cambio de estado en el sistema. Se representan por arcos dirigidos y etiquetas sobre ellos para indicar el evento que cambia el estado del sistema.

Del mismo modo que se trabajó para redes de colas y redes de Petri, se modela con Statechart el ejemplo del servidor Web en la Figura 3.11.



Figura 3.11: Statechart del servidor Web

En este gráfico se observa la representación del paralelismo a través de la división de cada componente entre líneas de punto. Además de los componentes modelados para las técnicas anteriores, se agrega el Generador y el Transductor.

Sobre cada transición se puede encontrar un acción o una secuencia que corresponde a **tiempo/acción (condición)**. La segunda posibilidad significa que pasado un tiempo se puede disparar una acción y además esa acción puede depender de una condición.

Para comprender el gráfico se explica el comportamiento de sus componentes:

**Generador:** produce los arribos de requerimientos cada  $\sigma_1$  unidades de tiempo, el

cual incrementa la cola de enlace entrante ( $inc\_ee$ ) pasado dicho tiempo.

**Enlace Entrante:** frente al incremento de la cola del enlace entrante ( $inc\_ee$ ) del Generador se pueden producir dos situaciones: (i) si el Enlace Entrante está en estado *pasivo* pasa a *activo*, (ii) en caso contrario permanece en *activo*. Cumplido el tiempo  $\sigma_2$  se debe evaluar que sucede con la cola  $Q$  del enlace entrante. Si ésta es mayor que uno, existen más requerimientos a atender y por tanto el enlace permanece en estado *activo*. Para el caso que sea uno, el estado pasa a *pasivo* ya que la cola está vacía. Como acción se decrementa la cola del enlace entrante ( $dec\_ee$ ) y se incrementa la cola del router ( $inc\_r$ ).

**Router:** frente al incremento de la cola, el router pasa al estado *activo*. Pasado un tiempo  $\sigma_3$  debe evaluarse si el requerimiento atendido proviene del enlace entrante o del saliente. En el caso que corresponda al enlace entrante debe pasar al procesador ( $dec\_r, inc\_proc$ ) y en el del enlace saliente debe salir ( $dec\_r$ ). Luego el estado pasa a *pasivo*.

**Procesador:** frente al incremento de la cola del procesador se pueden producir dos situaciones: (i) si el Procesador está en estado *pasivo* pasa a *activo*, (ii) en caso contrario permanece en *activo*. Cumplido el tiempo  $\sigma_4$  se debe evaluar que sucede con la cola  $Q$  del procesador. Si ésta es mayor que uno, existen más requerimientos a atender y por tanto el enlace permanece en estado *activo*. Para el caso que sea uno, el estado pasa a *pasivo* ya que la cola está vacía. Además se debe distinguir si el requerimiento fue atendido por el disco. En el caso afirmativo debe pasar al enlace saliente ( $dec\_proc, inc\_es$ ) y en caso negativo debe pasar al disco ( $dec\_proc, inc\_disco$ ). Luego el estado pasa a *pasivo*.

**Disco:** frente al incremento de la cola del disco ( $inc\_disco$ ) en el Procesador se pueden producir dos situaciones: (i) si el Disco está en estado *pasivo* pasa a *activo*, (ii) en caso contrario permanece en *activo*. Cumplido el tiempo  $\sigma_5$  se debe evaluar que sucede con la cola  $Q$  del disco. Si ésta es mayor que uno, existen más requerimientos a

atender y por tanto el enlace permanece en estado *activo*. Para el caso que sea uno, el estado pasa a *pasivo* ya que la cola está vacía. Como acción se decrementa la cola del disco (*dec\_disco*) y se incrementa la cola del procesador (*inc\_proc*).

**Enlace Saliente:** frente al incremento de la cola del enlace saliente (*inc\_disco*) del Procesador se pueden producir dos situaciones: (i) si el Enlace Saliente está en estado *pasivo* pasa a *activo*, (ii) en caso contrario permanece en *activo*. Cumplido el tiempo  $\sigma_6$  se debe evaluar que sucede con la cola  $Q$  del enlace saliente. Si ésta es mayor que uno, existen más requerimientos a atender y por tanto el enlace permanece en estado *activo*. Para el caso que sea uno, el estado pasa a *pasivo* ya que la cola está vacía. Como acción se decrementa la cola del enlace saliente (*dec\_es*) y se incrementa la cola del router (*inc\_r*).

**Transductor:** es el responsable de calcular las métricas de los requerimientos que finalizan el procesamiento. Ante una acción computar del Router, se operan los cálculos del trabajo atendido.

El diagrama Statechart es útil para diseñar modelos orientados a eventos con la posibilidad de determinar para cada componente, los estados y sus acciones en las transiciones. Con dichas acciones se obtienen las funciones de transición de estados sobre las cuales se elabora la semántica del modelo.

El aumento de la complejidad del problema se refleja en la representación por su característica gráfica, pero el beneficio que se destaca es la capacidad para identificar componentes atómicos y funciones de transición de estados que conducen a la elaboración de los algoritmos del formalismo. Por ello, es una herramienta adecuada para el diseño de modelos orientados a eventos.

## 3.6. Formalismo DEVS

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

DEVS es un formalismo de evento discreto que propone una teoría de modelado de sistemas para simulación. El formalismo permite especificar un objeto matemático, como entidad de un sistema, que se define por sus entradas, salidas, estados, una base de tiempo y funciones de transición que coordinan los cambios de estados y producen las salidas.

Cada objeto es un álgebra por la cual se define la estructura y sus reglas y es llamado modelo básico o atómico. Un modelo atómico puede ser interconectado con otros para la construcción de nuevos modelos llamados acoplados. De este modo se logra la representación de complejidad con las capacidades de modelado modular y jerárquico. La interconexión se implementa a través de puertos de entradas y salidas.

Un paradigma de modelado puede clasificarse respecto a (Wainer, 2003):

- **Base de tiempo:**

- **Tiempo continuo:** el tiempo evoluciona de forma continua, es un número real.
- **Tiempo discreto:** el tiempo avanza por saltos de un valor entero a otro.

■ **Valores de las variables descriptivas del modelo:**

- **Estados o eventos discretos:** las variables toman sus valores en un conjunto discreto.
- **Continuos:** las variables son números reales.
- **Mixtos:** las variables del sistema pueden ser continuas o discretas.

DEVS es un paradigma de tiempo continuo y de variables discretas, de modo que las entradas al sistema ocurren en momentos espaciados arbitrariamente con variables de estado que asumen un número discreto de valores definidos.

El proceso de eventos discretos se caracteriza por la habilidad para percibir un flujo de eventos de entrada como estímulos, atender su secuenciación y determinar el espacio de tiempo entre ellos.

Dado un conjunto arbitrario de eventos  $Z$  y una base de tiempo  $\mathbb{T} = \mathbb{R}_{[0,\infty]}$  definida en  $\{t | 0 \leq t \leq \infty\}$ .

**Definición 3.6.1** *Un evento en el tiempo se define como un par  $(t, e)$  tal que  $t \in \mathbb{T}$  y  $e \in Z^*$ , donde  $Z^*$  es el conjunto de todas las combinaciones finitas de eventos de  $Z$ , incluyendo el elemento sin evento  $\infty$ .*

Por ejemplo para dos eventos:

$$Z = \{(e_1, t_1); (e_2, t_2)\},$$

con  $Z = \{e_1, e_2\}$  y  $Z^* = \{\phi, e_1, e_2, e_1e_2\}$ .

**Definición 3.6.2** *Una trayectoria de eventos o segmento de eventos en el tiempo  $\omega : T \rightarrow Z^*$  es una descripción de los eventos en el tiempo sobre una línea con una dimensión de 0 a  $\infty$ .*

En la Figura 3.12 se representa el tiempo con una línea horizontal continua, en donde de izquierda a derecha se ubican los eventos en el tiempo.

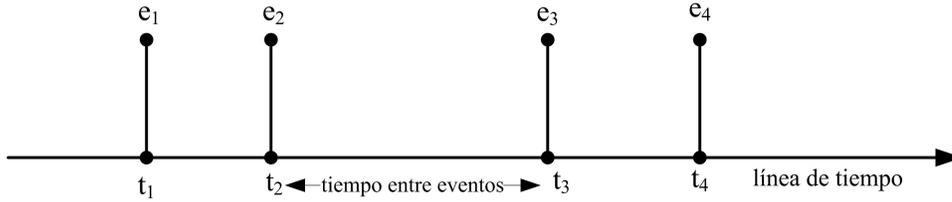


Figura 3.12: Representación de eventos en una línea de tiempo

En dicha línea se muestra la secuencia de eventos y la duración entre eventos, cuya trayectoria de eventos es:

$$\omega = \{\phi_{[0,t_1]}, (e_1, t_1), \phi_{[0,t_2]}, (e_2, t_2), \phi_{[0,t_3]}, (e_3, t_3), \phi_{[0,\infty]}\}.$$

### Definición 3.6.3 *Eventos concurrentes*

Si para un tiempo  $t$  arriba más de un evento a la vez se dice que los mismos son concurrentes y su segmento se representa como:

$$\omega = \{(e_1, t), (e_2, t), \dots, (e_n, t)\}.$$

### Definición 3.6.4 *Dominio de observación*

Un segmento puede ser definido en un intervalo de tiempo  $[t_{inicial}, t_{final}]$ . Por ello, el dominio de observación está acotado a:

$$\omega_{[t_{inicial}, t_{final}]} \rightarrow Z^*.$$

Por tanto, se modelan sistemas cuyos componentes son objetos. Cada objeto va a tener un comportamiento definido por funciones capaces de procesar una trayectoria de eventos (Definición 3.6.2).

Los eventos (Definición 3.6.1) arriban en tiempos no previstos. Frente a estos eventos se producen cambios de estados en las variables descriptivas del modelo. En algunos casos los objetos van a producir salidas, las cuales son expresadas como nuevos eventos en el tiempo.

Los objetos atómicos pueden vincularse con otros para formar un objeto acoplado y así sucesivamente hasta modelar el sistema en estudio. De este modo, se forma una constelación cuyas componentes son capaces de reaccionar al recibir un evento en un tiempo arbitrario y disparar otro como salida.

En la constelación puede suceder que dos o más eventos arriben al mismo instante de tiempo, por tanto el modelo debe ser capaz de atender eventos concurrentes (Definición 3.6.3).

Del segmento de tiempo se estudia un dominio de observación (Definición 3.6.4), sobre el cual se va a analizar su comportamiento.

### 3.6.1. Formalismo DEVS Clásico

La estructura del modelo se expresa en un formalismo, cuyo modelo atómico se presenta en la siguiente 7-upla y para la cual se definen sus conjuntos y funciones:

Un modelo DEVS atómico se representa por la estructura:

$$N = \langle X, Y, S, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

Donde:

- InputPorts es el conjunto de puertos de entradas a través de los cuales los eventos externos son recibidos.
- OutputPorts es el conjunto de puertos de salidas a través de los cuales los eventos externos son enviados.

- $X_p$  es el conjunto de valores permitidos en un puerto de entrada.
- $Y_p$  es el conjunto de valores permitidos en un puerto de salida.
- $X_n = \{(p, v) | (p \in \text{InputsPort}, v \in X_p)\}$  es el conjunto de puertos de entrada y valores de recepción permitidos desde un evento externo.
- $Y_n = \{(p, v) | (p \in \text{OutputPorts}, v \in Y_p)\}$  es el conjunto de puertos de salida y valores de emisión permitidos como evento de salida.
- $S$  es el conjunto de estados.
- $\delta_{int} = S \rightarrow S$  es la función de transición de estado interna, la cual especifica un nuevo estado del sistema luego de que el tiempo asignado por la función de avance del tiempo ha finalizado.
- $\delta_{ext} = Q \times X \rightarrow S$  es la función de transición de estado externa, la cual especifica cómo el sistema cambia de estado cuando se produce un evento de entrada.
  - $Q = \{(s, e) | (s \in S, 0 < e < ta(S))\}$  es el conjunto de estados totales.
  - $e$  es el tiempo transcurrido desde la última transición.
  - $s$  es el conjunto parcial de estados para una duración  $ta(s)$  si no ocurre un evento externo
- $ta = S \rightarrow R_{0, \infty}^+$  es la función de avance de tiempo. Asigna el tiempo en el cual va a permanecer un nuevo estado. Para casos especiales, 0 identifica un estado transitorio de duración cero,  $\infty$  identifica un estado permanente. El modelo permanece en el estado  $\infty$  hasta el arribo de un evento externo.

- $\lambda = QxS \rightarrow S$  es la función de salida y es la responsable de generar una salida antes de que la función de transición interna se ejecute.

El formalismo se centra en eventos que definen los cambios de valores de las variables para un instante de tiempo. En las funciones se definen como generar nuevos valores en las variables y el tiempo que las mismas deben permanecer en el nuevo estado. Un aspecto a destacar es que el tiempo entre eventos es variable.

De las diferentes variables del modelo, dos siempre están presentes, *phase* y *sigma* y su intervención consiste en:

- Ante la ausencia de evento externo el sistema permanece en *phase = passive* para un tiempo *sigma =  $\infty$* .
- La función de transición interna se ejecuta una vez que finaliza el tiempo *sigma* para el estado *phase*.
- Para planificar la nueva *phase* la función de transición externa toma en cuenta el estado corriente, el evento que ingresa al puerto de entrada y su valor, así como el tiempo que ha transcurrido en el estado actual.
- La función de avance del tiempo asigna el valor a *sigma* cuando se lo requiere.

Para comenzar a comprender el comportamiento de un modelo atómico se hace referencia a la Figura 3.13. En la misma se observa en el centro un conjunto de estados, cuyos valores pueden ser modificados de acuerdo a las diferentes funciones.

Cada componente del modelo se comunica con el entorno por puertos, a través de los cuales recibe y genera segmentos de eventos.

Las entradas son percibidas y atendidas por la función de transición externa. Ante un cambio de estado, es la función de avance de tiempo la que asigna el valor de permanencia es ese estado.

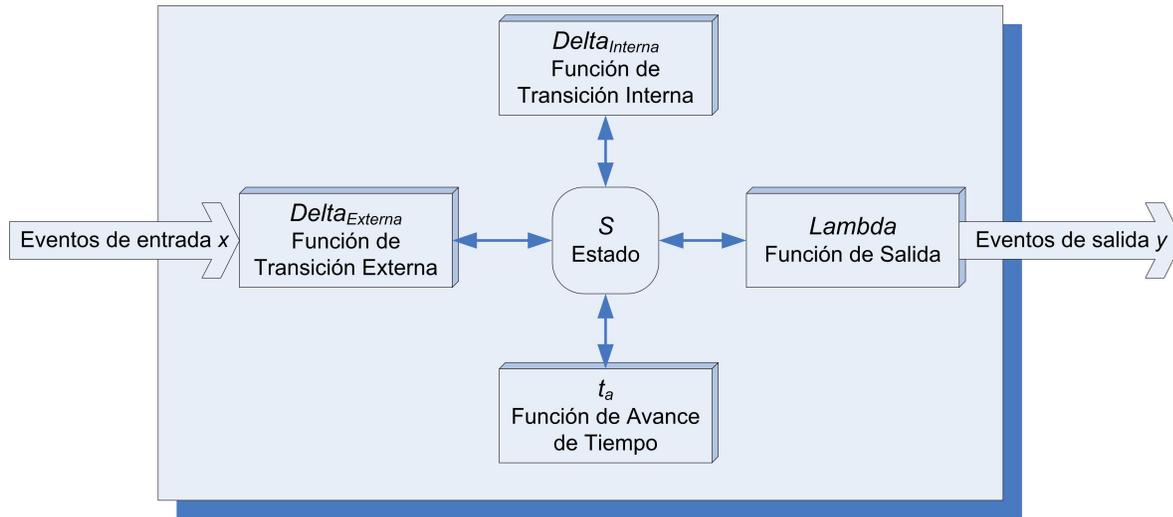


Figura 3.13: Modelo Atómico

Finalizado el tiempo asignado a un estado, se produce una salida como está previsto en la función de salida y luego es la función de transición interna la que asume el control y ejecuta su acción.

La semántica de un modelo atómico se basa en el siguiente razonamiento:

- Los cambios de estados  $S$  se producen a través de las funciones de transición.
- A cada estado se le asocia un tiempo de permanencia calculado por la función de avance del tiempo  $t_a$ .
- Si no ocurren eventos externos, los estados pueden cambiar ejecutando la función de transición interna  $\delta_{int}$ . Antes de cada transición interna se puede generar un evento de salida con la ejecución de la función de salida  $\lambda$ .
- Frente al arribo de un evento de entrada, la función de transición de estado externa  $\delta_{ext}$  es quien determina el nuevo estado sobre la base del estado corriente, el tiempo transcurrido  $e$  y la nueva entrada. Una característica a destacar es que la función  $\delta_{ext}$  no produce salida. Frente al cambio de estado  $e$  vuelve a cero.

- Los estados pueden ser transitorios o permanentes. Para el caso transitorio se encuentra activo por el tiempo asignado por la función  $ta$  y para el permanente se encuentra pasivo por un tiempo infinito. Si el estado es pasivo el modelo puede reactivarse sólo por la ocurrencia de un evento externo.

Para generalizar la comprensión de la semántica de un modelo DEVS atómico se plantea la función de transición de estado total  $\delta = Q \times Z \rightarrow Q$  como:

Para  $q = (s, e) \in Q$  y  $z \in Z$ :

$$\delta((s, e), z) = \begin{cases} (\delta_{ext}(q, z), 0) \text{ para } z \in X, \delta_{ext}(q, z) \text{ está definida} \\ (s, e) \text{ para } z \in X, \delta_{ext}(q, z) \text{ no está definida} \\ (\delta_{int}(s), 0) \text{ para } e = ta(s), z = \lambda(s), \delta_{int} \text{ está definida} \\ (error, e) \text{ en otro caso} \end{cases}$$

### 3.6.2. Modelos Acoplados

Los modelos atómicos pueden ser vinculados para formar un modelo acoplado, es decir se conectan modelos componentes para constituir un nuevo modelo.

En la Figura 3.14 se representa el modelo acoplado, que contiene un conjunto de componentes, un conjunto de puertos de entrada a través de los cuales se reciben los eventos externos, un conjunto de puertos internos y un conjunto de puertos de salida a través de los cuales se envían los eventos externos.

A través del External Output Coupling (EOC) se conectan los puertos de entrada del modelo acoplado con los de entrada de sus componentes, las conexiones internas del modelo se realizan por los Internal Coupling (IC) y las salidas de los componentes se conectan a las del modelo acoplado por los External Output Coupling (EOC).

Un modelo DEVS acoplado se representa por la estructura:

$$Coupled = \langle X, Y, D, \{M_i \text{ con } i \in D\}, EIC, IC, EOC, Select \rangle$$

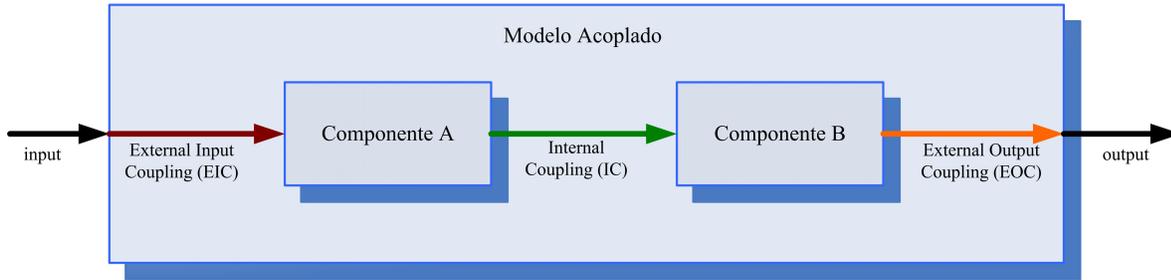


Figura 3.14: Modelo Acoplado

Donde:

- $X = \{(p, v)\}$ ;  $p \in InputPort$  y  $v \in X_p$ , siendo  $InputPort$  el conjunto de puertos de entrada y  $X_p$  los valores de entrada permitidos para el puerto  $p$ .
- $Y = \{(p, v)\}$ ;  $p \in OutputPort$  y  $v \in Y_p$ , siendo  $OutputPort$  el conjunto de puertos de salida e  $Y_p$  los valores de salida para el puerto  $p$ .
- $D$  es el conjunto de nombres de sub-componentes. Para cada  $i \in D$ :
- $M_i = \langle X_i, Y_i, S_i, \delta_{int_i}, \delta_{ext_i}, \lambda_i, ta_i \rangle$  es un modelo DEVS.
- $EIC \subseteq \{((M, ipN)(d, ipd))\}$  con  $ipN \in InputPort$ ,  $d \in D$  e  $ipd \in InputPort_d$ .  
External Input Coupling: es el conjunto de enlaces de entrada que conecta las entradas de un modelo acoplado a una o más entradas de sus componentes.
- $IC \subseteq \{((a, opa)(b, ipb))\}$  con  $a, b \in D$ ,  $opa \in OutputPort_a$ ,  $ipb \in InputPort_b$  y  $a \neq b$ .  
Internal Coupling: es el conjunto de enlaces internos que conectan puertos de salidas con entradas de componentes del modelo acoplado.
- $EOC \subseteq \{((d, opd)(M, opM))\}$  con  $opM \in OutputPort$ ,  $d \in D$  y  $opd \in OutputPort_d$ .  
External Output Coupling: es el conjunto de enlaces de salidas que conecta las salidas de uno o más componentes a la salida del modelo acoplado.

- $Select : 2^D - \{\} \rightarrow D$  es la función que evita la ejecución de eventos concurrentes.

Un modelo acoplado puede ser expresado como un nuevo modelo atómico y de este modo ser empleado como componente de un modelo acoplado de nivel superior, lo que demuestra que el formalismo DEVS es cerrado bajo acoplamiento, condición necesaria para la construcción jerárquica.

Es así que un modelo acoplado se transforma en cimiento de un nuevo modelo con el fin de construir una estructura jerárquica y modular.

### 3.6.3. Formalismo DEVS Paralelo

Los modelos pueden presentar eventos concurrentes, es decir varios componentes puedan activarse a la vez y enviar las salidas producidas a otros componentes.

De este modo el receptor se enfrenta a una o más entradas y por tanto debe estar preparado para interpretarlas. En este caso se produce una modificación al modelo DEVS tradicional para incorporar el concepto de DEVS paralelo.

Respecto a DEVS clásico, el paralelo presenta las siguientes capacidades: (i) los modelos operan con bags de entrada y de salida, es decir tienen la capacidad de recibir múltiples eventos de entrada y enviar más de un evento de salida y (ii) se incorpora una nueva función llamada de transición de confluencia, la misma tiene la tarea de decidir el próximo estado para el caso de colisión entre un evento externo e interno.

Un modelo DEVS Paralelo se representa por la estructura:

$$P = \langle X_p^b, Y_p^b, S, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

Donde:

- $X_p^b = \{((p, v) | (p \in \text{InputPorts}, v \in X_p))\}$  es el conjunto de puertos y múltiples eventos de entrada permitidos.

- $Y_p^b = \{(p, v) | (p \in \text{OutputPorts}, v \in Y_p)\}$  es el conjunto de puertos y múltiples eventos de salida permitidos.
- $S$  es el conjunto de estados internos.
- $\delta_{int} : S \rightarrow S$  es la función de transición de estado interna.
- $\delta_{ext} : Qx X_p^b \rightarrow S$  es la función de transición de estado externa.
  - con  $Q = \{(s, e) | (s \in S, 0 < e < ta(S))\}$  es el conjunto de estados totales.
- $\delta_{conf} : Qx X^b \rightarrow S$  es la función de transición de confluencia.
- $\lambda : S \rightarrow Y^b$  es la función de salida.
- $ta : S \rightarrow R_{0, \infty}^+$  es la función de avance de tiempo.

Por tanto, en DEVS paralelo se manejan conjuntos de entradas y salidas (bags). Frente a la concurrencia pueden suceder casos en donde eventos internos y externos suceden a la vez, por ello se incorpora la función de transición de confluencia que organiza la secuencia de ejecución.

### 3.6.4. Protocolo de Simulación DEVS Paralelo

En la Figura 3.15 se muestra el ciclo básico para llevar adelante el proceso de simulación DEVS (Zeigler y Sarjoughian, 2003). Para comprender este proceso se detallan las variables a utilizar:

- $s$ : estado.
- $s_{inicial}$ : valor asignado por el modelador al estado inicial.
- $t$ : tiempo de simulación.

- $tL$ : tiempo del último evento.
- $tN$ : tiempo del próximo evento.

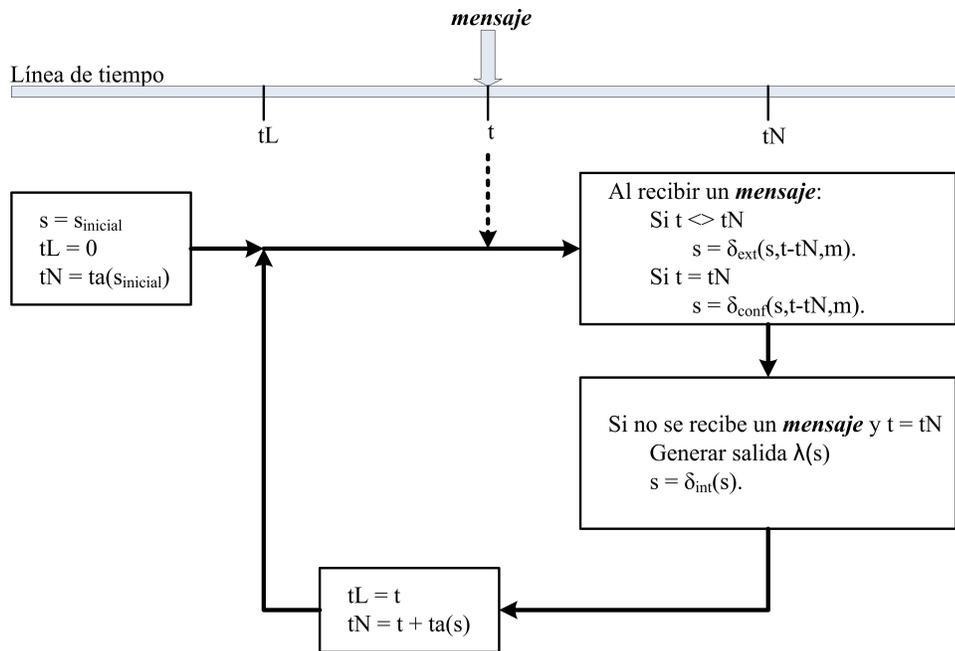


Figura 3.15: Proceso de Simulación DEVS

El proceso comienza con la inicialización de las variables  $s$  en el estado inicial propuesto por el modelador,  $tL$  en cero y  $tN$  en el valor asignado por la función de avance de tiempo para el estado  $S_{inicial}$ .

En el tiempo de simulación puede suceder el arribo de un evento, lo cual se indica con la llegada de un mensaje en un tiempo  $t$  determinado. Frente al arribo de este evento se inicia un ciclo que consiste en comparar el tiempo actual de simulación  $t$  respecto al tiempo  $tN$ . Si no existe coincidencia se ejecuta la función de transición externa, de lo contrario se realiza la función de transición de confluencia.

Es decir, si al arribo de un mensaje el tiempo  $t$  es distinto de  $tN$  no se cumplió el tiempo asignado al estado corriente por tanto es la función de transición externa la

responsable de decidir el cambio de estado. Si es igual significa la llegada de un mensaje al mismo tiempo que finaliza el estado actual y para ello es la función de confluencia quien se encarga de decidir el comportamiento futuro.

El ciclo también puede iniciarse por la finalización del tiempo asignado al estado corriente sin existir un mensaje entrante. Para este caso se genera un evento de salida por parte de la función de salida y se produce el cambio de estado desde la función de transición interna.

Para los dos casos anteriores de ciclo se actualizan las variables  $tL$  con el tiempo actual de simulación y  $tN$  con el tiempo actual de simulación más el valor asignado al nuevo estado por la función de avance de tiempo.

En este proceso de simulación cada modelo atómico tiene asociado un simulador y para sincronizar los ciclos existe un coordinador de los diferentes simuladores. Para ello el protocolo de simulación DEVS construye un árbol AVL como se muestra en la Figura 3.16.

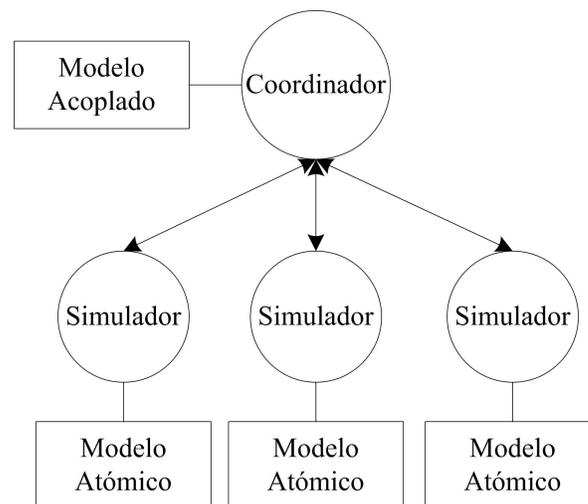


Figura 3.16: Estructura AVL del Protocolo DEVS

Esta estructura contiene como elementos hojas los modelos atómicos. Con esta estructura se construye una cola desde los elementos hojas y se almacenan datos claves

como nombre del elemento y tiempo del próximo evento. El coordinador sincroniza el ciclo de simulación (Zeigler (2003)) a través de los siguientes pasos:

1. Buscar el tiempo menor en la cola y actualizar el reloj de simulación.
2. Seleccionar todos los modelos hojas cuyo tiempo del próximo evento coincide con el reloj. Con los elementos seleccionados formar el conjunto de modelos inminentes llamado I.
3. Ejecutar las funciones de salidas de los modelos del conjunto I y enviar los mensajes de salida a los puertos de entrada correspondientes. Los mensajes enviados son recolectados en un conjunto de múltiples eventos por los puertos de entrada.
4. Seleccionar todos los modelos que contienen un conjunto de múltiples eventos de mensajes esperando en sus entradas y formar el conjunto M.
5. Encontrar la intersección de los elementos de los conjuntos I y M, al cual se lo llama conjunto confluencia. Para todos los modelos del conjunto confluencia ejecutar la función de transición de confluencia, que primero atiende las transiciones internas y luego las externas.
6. Para el resto de los elemento del conjunto I que no pertenecen a la intersección, ejecutar la función de transición interna asociada a cada modelo.
7. Para el resto de los elementos del conjunto M que no pertenecen a la intersección, ejecutar la función de transición externa asociada a cada modelo.

Una vez ejecutados los pasos anteriores se actualizan los próximos eventos de tiempo de acuerdo a las funciones de avance del tiempo asociadas a cada modelo. Se reconstruye la cola y se repiten los pasos 1 a 7 hasta que no existan modelos en el conjunto I o se produzca una condición de fin de la simulación.

Para especificar la relación entre coordinador y simulador se presenta el intercambio de mensajes en la Figura 3.17.

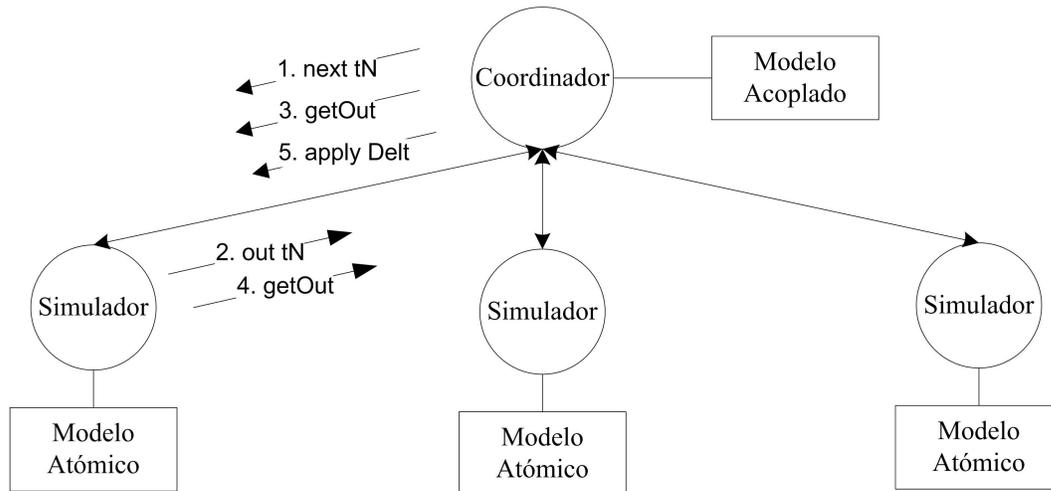


Figura 3.17: Secuencia entre coordinador y simuladores

El coordinador envía a cada simulador el next tN (1) solicitando su tN. Los simuladores le responden sus valores de tNs en el mensaje out tN (2).

De los tNs recibidos, el coordinador obtiene el mínimo y lo informa a los simuladores a través de un mensaje getOut (3).

Cada simulador debe verificar si su tN coincide con el tN global, en caso afirmativo es parte del conjunto inminente I y debe enviar su salida al coordinador en un mensaje getOut (4).

El coordinador distribuye las salidas a los simuladores con el mensaje applyDelt (5).

Para generalizar en una estructura jerárquica de mayor complejidad se propone en la Figura 3.18 un caso al cual se le asocia la estructura de simulador.

Cada modelo atómico tiene asociado su simulador (clase Simulator). Los modelos acoplados son administrados por un coordinador (clase Coordinator).

Los simuladores se basan en un protocolo de mensajes que junto a los coordinadores implementan la simulación. El coordinador utiliza la estructura de acoplamientos para

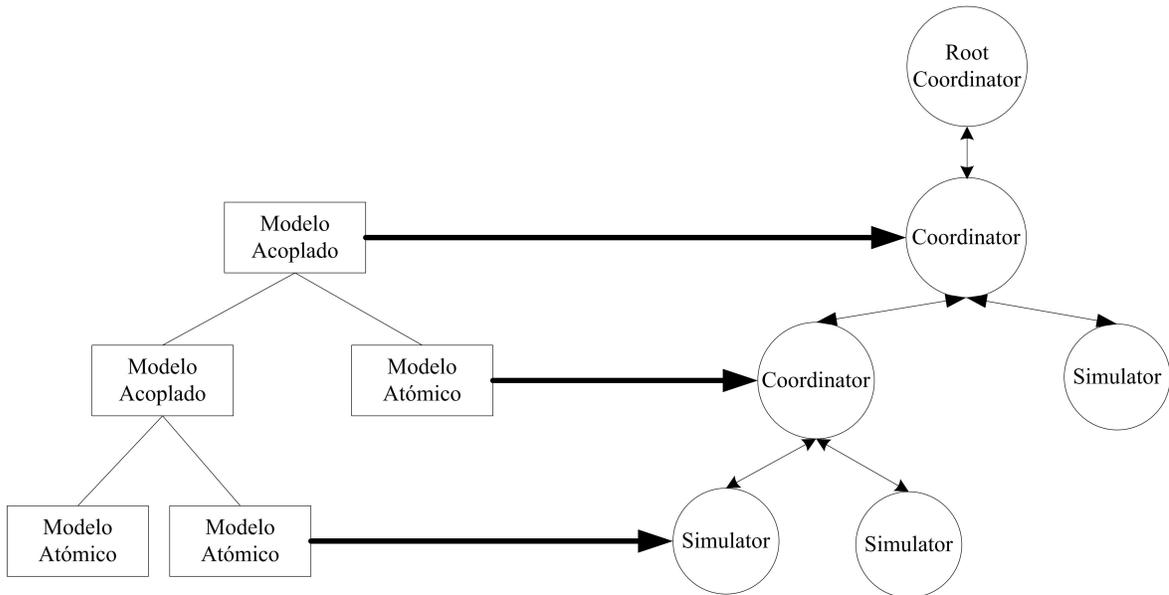


Figura 3.18: Estructura jerárquica y su modelo de simulador asociado

distribuir las salidas hacia los simuladores correspondientes.

La asignación de simuladores y coordinadores se basa en el modelo jerárquico de componentes, cuya raíz es la clase Root-Coordinator.

La clase raíz es la responsable de implementar los ciclos de simulación y de este modo sincronizar los simuladores subordinados.

### 3.6.5. Formalismo DEVS Estocástico

Un modelo DEVS es determinístico si las funciones  $\delta_{int}$ ,  $\delta_{ext}$ ,  $\lambda$  y  $ta$  son determinísticas, en otro caso el modelo DEVS es no determinístico.

Castro y otros (2010) proponen el STDEVS (STochastic DEVS) como extensión del formalismo DEVS clásico. STDEVS es un framework que proporciona las especificaciones para modelado y simulación de sistemas de eventos discretos con comportamiento estocástico.

Se basa en el uso de espacios de probabilidades y su dinámica interna consiste en

transiciones de estados como experimentos aleatorios.

La estructura de un modelo atómico STDEVS es la siguiente:

$$M_{STDEVS} = \langle X_p, Y_p, S, \zeta_{int}, \zeta_{ext}, P_{int}, P_{ext}, \lambda, ta \rangle$$

Dado

- El espacio muestral de experimentos aleatorios  $S$ , el conjunto que incluye todas las salidas posibles de un experimento.
- Un sigma-álgebra  $\mathcal{F}$ : un espacio de eventos del espacio muestral  $S$ , es decir un conjunto no vacío de subconjuntos de  $S$ , que cumple con las siguientes propiedades:
  - El conjunto vacío está en  $\mathcal{F}$ .
  - Si  $F$  está en  $\mathcal{F}$ , también está su complemento.
  - Si  $F_1, F_2$  y  $F_3$  es una sucesión de elementos de  $\mathcal{F}$ , entonces la unión de todos ellos también está en  $\mathcal{F}$ .
- Un sigma-álgebra particular de  $S$  llamado  $2^S$ : es una colección de todos los subconjuntos del espacio muestral  $S$ .
- Un espacio medible identificado por el par  $(S, \mathcal{F})$ : espacio muestral  $S$  y sigma-álgebra de subconjuntos de  $S$ .
- $P(\mathcal{F})$  es la asignación de una probabilidad  $P$  sobre cada miembro del sigma-álgebra  $\mathcal{F}$ , tal que  $P(F) \geq 0 \forall F \in \mathcal{F}$ .
- Un espacio de probabilidad  $(S, \mathcal{F}, P)$  consistente en un espacio muestral  $S$ , un sigma-álgebra  $\mathcal{F}$  de subconjuntos de  $S$  y una medida de probabilidad  $P$  definida para todos los miembros de  $\mathcal{F}$ . Por tanto:

- $\forall F \in \mathcal{F}$ ,  $P(F)$  expresa la probabilidad de que un experimento produzca una muestra  $s \in F \subseteq S$ .

Donde:

- $X$ ,  $Y$  y  $S$  se corresponden con los conjuntos de entradas, salidas y estados respectivamente definidos en el formalismo DEVS clásico
- $\lambda$  y  $ta$  se corresponden con las funciones de salida y de avance del tiempo definidas en el formalismo DEVS clásico.
- $\zeta_{int} : S \rightarrow 2^S$  es la función que asigna una colección de conjuntos  $\zeta_{int}(s) \subseteq 2^S$  a cada estado  $s$ .
  - Dado un estado  $s$ , la colección  $\zeta_{int}(s)$  contiene todos los subconjuntos de  $S$  que el próximo estado podría alcanzar de acuerdo a una probabilidad dada por la función  $P_{int}$ .
- $\zeta_{ext} : SxR_0^+xX \rightarrow 2^S$  es la función que asigna una colección de conjuntos  $\zeta_{ext}(s, e, x) \subseteq 2^S$  a cada triplo  $(s, e, x)$ .
  - Dado un estado  $s$  y un tiempo transcurrido  $e$ , si un evento  $x$  arriba,  $\zeta_{ext}(s)$  contiene todos los subconjuntos de  $S$  que el próximo estado podría alcanzar de acuerdo a una probabilidad dada por la función  $P_{ext}$ .
- $P_{int} : Sx2^S \rightarrow [0, 1]$  es la función de probabilidad y calcula la probabilidad de transición interna.
- $P_{ext} : SxR_0^+xXx2^S \rightarrow [0, 1]$  es la función de probabilidad y calcula la probabilidad de transición externa.

### 3.6.6. Formalismo DEVS Paralelo y Estocástico

Castro y otros (2010) demuestran la implementación del modelo STDEVS a través del uso del modelo DEVS tradicional y la incorporación de funciones aleatorias en las funciones de transición.

En la tesis se propone la implementación de la siguiente estructura DEVS paralela y estocástica del modelo atómico:

$$P = \langle X_p, Y_p, S, \delta_{int}, \delta_{ext}, \delta_{conf}, P_{\lambda, out}, \lambda_{p, out}, ta \rangle$$

- $X_p = \{(p, v) | (p \in \text{InputPorts}, v \in X_p)\}$  es el conjunto de puertos de entrada  $p$  y de múltiples valores de eventos de entrada permitidos.
- $Y_p = \{(p, v) | (p \in \text{OutputPorts}, v \in Y_p)\}$  es el conjunto de puertos de salida  $p$  y de múltiples valores de eventos de salida permitidos.
- $S$  es el conjunto de estados internos.
- $\delta_{int} : S \rightarrow S$  es la función de transición de estado interna.
- $\delta_{ext} : Q \times X_p \rightarrow S$  es la función de transición de estado externa.
  - $con Q = \{(s, e) | (s \in S, 0 < e < ta(S))\}$  es el conjunto de estados totales.
- $\delta_{conf} : Q \times X \rightarrow S$  es la función de transición de confluencia.
- $\lambda_{p, out} : S \times P \rightarrow Y, r \approx U(0, 1)$  es la función de salida estocástica, dependiendo del conjunto de probabilidades  $P$  se va a seleccionar aleatoriamente el puerto de salida.
- $T$  es el conjunto de transacciones del sitio (search, browse, visit homepage, add cart).

- $P_{\lambda,out}(t, p, t') : TxP \rightarrow T$  es la función de probabilidades que asigna la probabilidad  $p$  de pasar de una transacción  $i$  a la próxima transacción  $j$ .
- $ta : S \rightarrow R_{0,\infty}^+$  es la función de avance del tiempo, su valor es el resultado de una distribución de probabilidad  $Dist_{ta}$ .

En la estructura propuesta se implementa el comportamiento estocástico a través de la función de salida estocástica. En primer lugar se cuenta con una función de probabilidades  $P_{\lambda,out}(t, p, t')$  que contiene triplos formados por la transacción actual, una probabilidad de transición y la transacción siguiente de acuerdo a dicha probabilidad.

En la función de salida la selección del puerto es aleatoria, de modo que se genera un número aleatorio  $r$  y se selecciona la próxima transacción de acuerdo a  $r$  y a la función de probabilidad.

### 3.7. Framework DEVS para Modelización de Transacciones Electrónicas

El framework DEVS se define por tres elementos básicos: sistema real, modelo y simulador (Palaniappan y otros, 2006), los cuales se presentan en la Figura 3.19.

El sistema real se considera como fuente de datos, del cual se abstraen los componentes y relaciones que luego serán elementos para la construcción y validación del modelo. Este sistema fuente puede ser existente o propuesto.

El modelo es el conjunto de instrucciones que definen la estructura y cuyo comportamiento debe ser similar al sistema fuente. El simulador es el responsable de la ejecución del modelo de acuerdo a un protocolo formal abstracto.

De la observación del sistema real se obtienen datos que posibilitan la construcción del modelo y son referencia para la validación del mismo. Además se abstrae un Marco

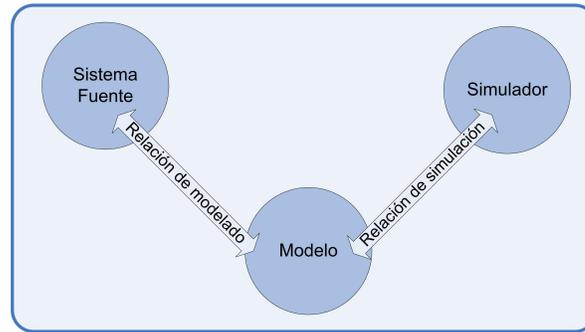


Figura 3.19: Elementos básicos del framework DEVS

Experimental consistente en un Generador y un Transductor. El Generador produce un segmento de eventos de entrada al modelo, mientras que el Transductor computa el segmento de eventos de salida para el cálculo de las métricas del sistema.

Respecto a la relación de simulación se destaca la independencia entre modelo y simulador, de modo que el conjunto de instrucciones que conforman el modelo se perfeccionan para ser ejecutadas de acuerdo a un protocolo abstracto de simulación.

Para comenzar a construir el framework DEVS de procesos de comercio electrónico se identifican sus componentes. Sobre la base del diagrama de estructura de la Figura 3.20 se esquematiza el modelo de proceso de comercio electrónico de acuerdo a una metodología top-down.

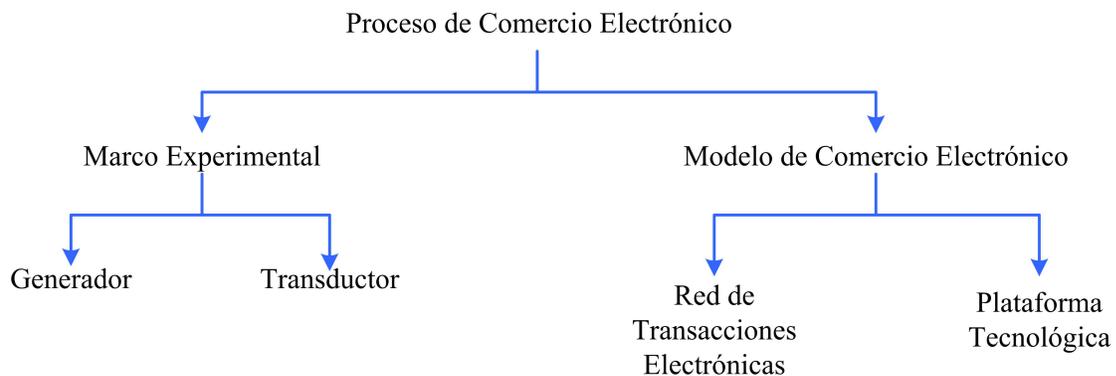


Figura 3.20: Diagrama de Estructura del Proceso de Comercio Electrónico.

La raíz es el Proceso de Comercio Electrónico y sus estructuras principales son el Marco Experimental y el Modelo de Comercio Electrónico.

El Marco Experimental consta de un Generador y un Transductor, mientras que el Modelo de Comercio Electrónico se compone de los modelos de Transacciones Electrónicas y de la Plataforma Tecnológica.

La estructura Transacciones Electrónicas consiste en un modelo funcional de proceso de negocios que comprende las operaciones que los clientes pueden realizar en el sitio y las necesarias para implementar el proceso de comercio electrónico. De modo que las transacciones se vinculan en una red de transiciones y conforman los diferentes caminos que un cliente puede tomar en su navegación en un sitio web de comercio. Cada camino particular desde el ingreso de un cliente hasta su partida se corresponde con una sesión.

Por otro lado, el modelo de la Plataforma Tecnológica describe los recursos informáticos y de comunicaciones que soportan el proceso de comercio electrónico.

Para elaborar los modelos DEVS se utiliza, en otro sentido, una metodología bottom-up, de modo que partiendo desde los nodos hojas se procede a identificar cada modelo atómico y en cada nodo padre se establecen los acoplamientos.

Las relaciones entre el marco experimental y el modelo se representan en la Figura 3.21. Se observan flechas que ingresan a un componente o salen del mismo. Las mismas se corresponden con puertos de entrada y salida que se detallarán en las siguientes secciones.

El Generador produce los ingresos al sistema de acuerdo con un modelo de carga de trabajo. Este modelo se obtiene a partir de un análisis de la frecuencia de arribos de clientes al sitio y por sus características no determinísticas se representa por una distribución de probabilidad.

Cada vez que el Generador produce un arribo, éste es ingresado al Modelo y además informado al Transductor. El Transductor almacena información del tiempo de llegada

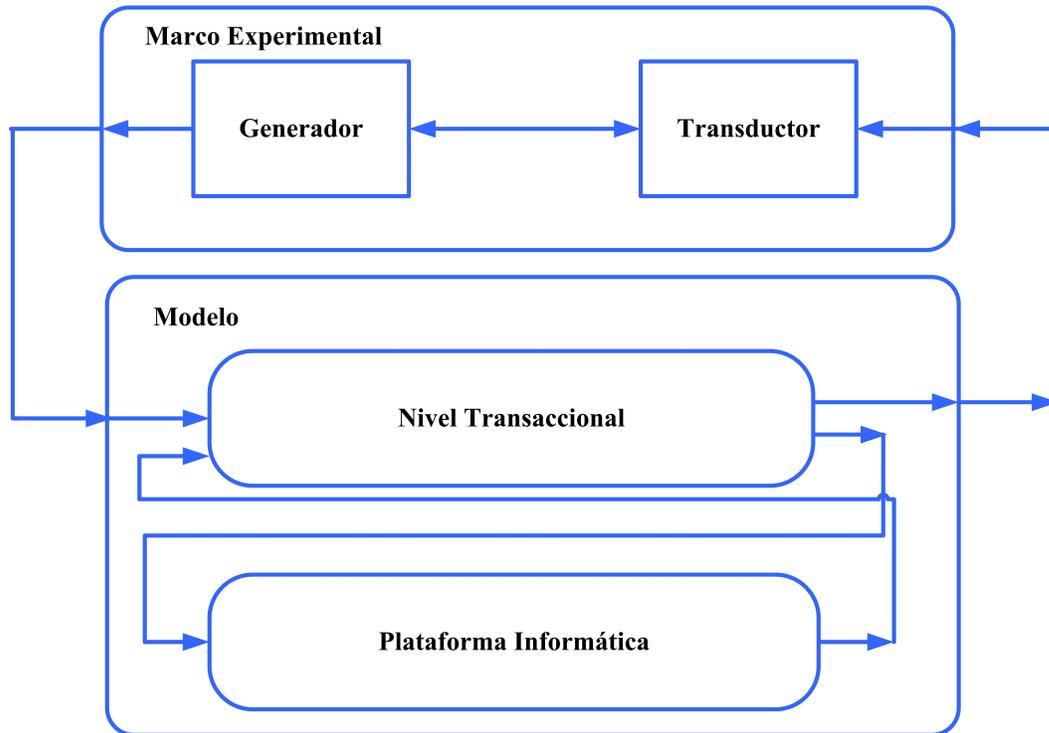


Figura 3.21: Componentes del Framework DEVS

de cada arribo para calcular la duración de una sesión al confrontarla con su tiempo de salida, además acumula la cantidad de sesiones finalizadas para obtener la productividad del sistema.

Cuando un arribo que ingresa al modelo es recibido por una transacción electrónica, pasa el control a la Plataforma Informática para su ejecución. En esta Plataforma la transacción es atendida por un recurso informático de acuerdo a un tiempo de servicio. Una vez completado el servicio, se devuelve el control al nivel transaccional para seguir camino a la próxima transacción. Si es una transacción de salida el control es derivado al Transductor.

Pueden existir varias transacciones de entradas y de salidas. Esto es, un cliente puede ingresar por la página inicial del sitio o por un link obtenido en un buscador. Del mismo modo puede salir luego de concretar la compra o abandonar el sitio en cualquiera

de sus transacciones.

En las siguientes secciones se desarrollan los componentes atómicos y los acoplamientos del modelo de comercio electrónico.

## 3.8. Marco Experimental

El Marco Experimental es un modelo acoplado de los módulos Generador y Transductor. El Generador es un modelo atómico capaz de proveer las entradas al sistema y el Transductor es responsable de monitorizar la ejecución del proceso de comercio electrónico y observar las salidas.

### 3.8.1. Generador

El generador es un componente del marco experimental responsable de producir los arribos de requerimientos al sistema, su especificación gráfica se muestra en la Figura 3.22. EL modelo atómico Generador consta de un puerto de entrada *stop* y un puerto de salida *out*, sus estados posibles son *active* y *passive*. En el estado inicial se encuentra en *active*.

Su semántica consiste en: el modelo permanece en el estado *active* por una duración equivalente al tiempo entre arribos de requerimientos; luego de este período la función de salida genera un arribo al sistema llamado *reqses<sub>i</sub>* y la función de transición interna reinicia el estado en *active* por un nuevo período.

Si ingresa el evento externo *end* por el puerto *stop*, la función de transición externa produce un cambio al estado *passive* sin ejecutar salida.

El modelo DEVS del Generador es la estructura:

$$\text{Generador} = \langle X_g, Y_g, S_g, \delta_{intg}, \delta_{extg}, ta_g, \lambda_g \rangle$$

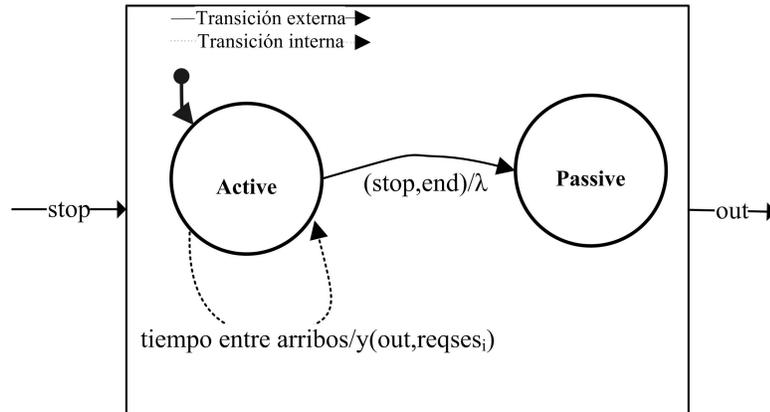


Figura 3.22: Diagrama Statechart del Generador

### Conjuntos de puertos y valores permitidos

$InputPort = \{stop\}$  conjunto de puertos de entrada.

$OutputPort = \{out\}$  conjunto de puertos de salida.

$X_{stop} = \{end\}$  es el conjunto de valores permitidos para el puerto de entrada  $stop$ .

$Y_{out} = \{reqses_i\}$  con  $i = 1..n$ , es el conjunto de valores permitidos para el puerto de salida  $out$ .

### Conjuntos de eventos de entrada y salida

$X_g = \{(stop, end)\} \mid stop \in InputPort \text{ y } end \in X_{stop}$

$Y_g = \{(out, reqses_i)\} \mid out \in OutputPort \text{ y } reqses_i \in Y_{out}$

### Conjunto de estados

$S_g = \{active, passive\}$

### Funciones

Función de transición de estados interna:

$\delta_{int}(active) = active$

Función de transición de estados externa:

$\delta_{ext}(active, e, (stop, end)) = passive$

Función de salida:

$$\lambda(active) = \{(out, reqses_i)\}$$

Función de avance del tiempo:

$$ta: \begin{cases} ta(active) = \text{tiempo entre arribos} \\ ta(passive) = \infty \end{cases}$$

### 3.8.2. Transductor

El Transductor es el segundo componente del Marco Experimental y su objetivo es registrar los requerimientos arribados y atendidos para computar las métricas del tiempo de respuesta y la velocidad de procesamiento del sistema. Su diagrama Statechart se muestra en la Figura 3.23.

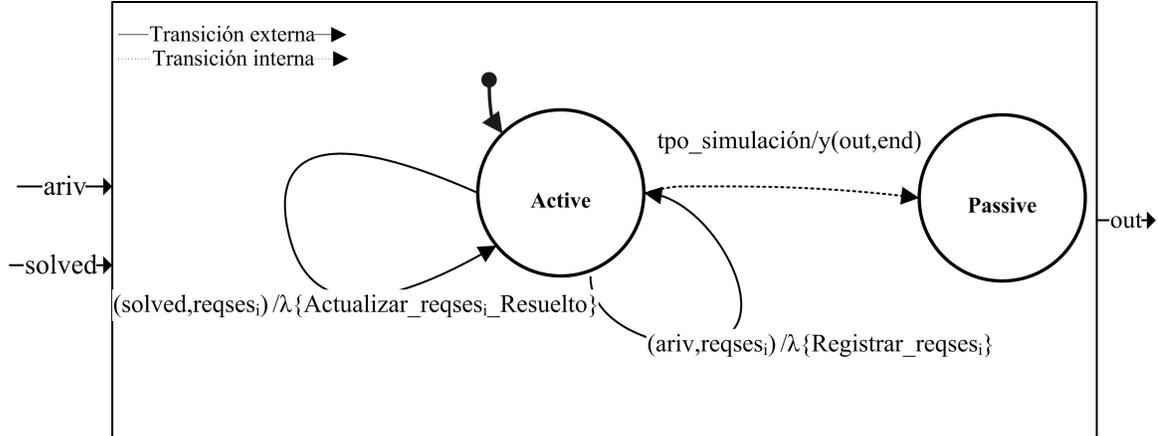


Figura 3.23: Diagrama Statechart del Transductor

Los puertos de entrada son *ariv* para recibir cada requerimiento generado y *solved* para ingresar el requerimiento que finaliza su navegación.

Su semántica consiste en: dado un estado inicial *active*, permanece en el mismo por un tiempo determinístico correspondiente al tiempo establecido para la simulación. Cumplido el tiempo de simulación la función de salida genera un evento *end* por el

puerto *out* como indicador de fin de simulación y la función de transición interna cambia al estado *passive*.

La función de transición externa se puede producir por dos tipos de eventos: (i) el arribo del valor  $reqses_i$  por el puerto *ariv* o (ii) el arribo de  $reqses_i$  por el puerto *solved*. Para el primer caso se registra el requerimiento como una nueva sesión, con un número que la identifica, mientras que para el segundo se computa una sesión finalizada. En ambos casos permanece en el estado *active*.

El modelo DEVS del Transductor es la estructura:

$$\text{Transductor} = \langle X_t, Y_t, S_t, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

### Conjuntos de Puertos y valores permitidos

$InputPorts = \{ariv, solved\}$  conjunto de puertos de entrada.

$OutputPort = \{out\}$  conjunto de puertos de salida.

$X_{ariv} = \{reqses_i\}$  con  $i = 1..n$ , es el conjunto de valores permitidos para el puerto de entrada *ariv*.

$X_{solved} = \{reqses_i\}$  con  $i = 1..n$ , es el conjunto de valores permitidos para el puerto de entrada *solved*.

$Y_{out} = \{end\}$  es el conjunto de valores permitidos para el puerto de salida *out*.

### Conjuntos de eventos de entrada y salida

$X_t = \{(ariv, reqses_i), (solved, reqses_i)\} \mid ariv, solved \in InputPort, reqses_i \in X_{ariv}, reqses_i \in X_{solved}$

$Y_t = \{(out, end)\} \mid out \in OutputPort \text{ y } end \in Y_{out}$

### Conjunto de estados

$S_t = \{active, passive\}$

## Funciones

Función de transición de estados interna:

$$\delta_{int}(active) = passive$$

Función de transición de estados externa:

$$\delta_{ext} : \begin{cases} \delta_{ext}(active, e, (ariv, reqses_i)) = active \\ \delta_{ext}(active, e, (solved, reqses_i)) = active \end{cases}$$

Función de transición de salida:

$$\lambda(active) = \{(out, end)\}$$

Función de avance del tiempo:

$$ta : \begin{cases} ta(active) = \text{tiempo de simulación} \\ ta(passive) = \infty \end{cases}$$

### 3.8.3. Modelo Acoplado del Marco Experimental

El Marco Experimental es un modelo acoplado de los modelos atómicos Generador y Transductor. Su diseño refleja las condiciones de experimentación sobre el modelo del sistema. En la Figura 3.24 se grafican sus componentes y acoplamientos.

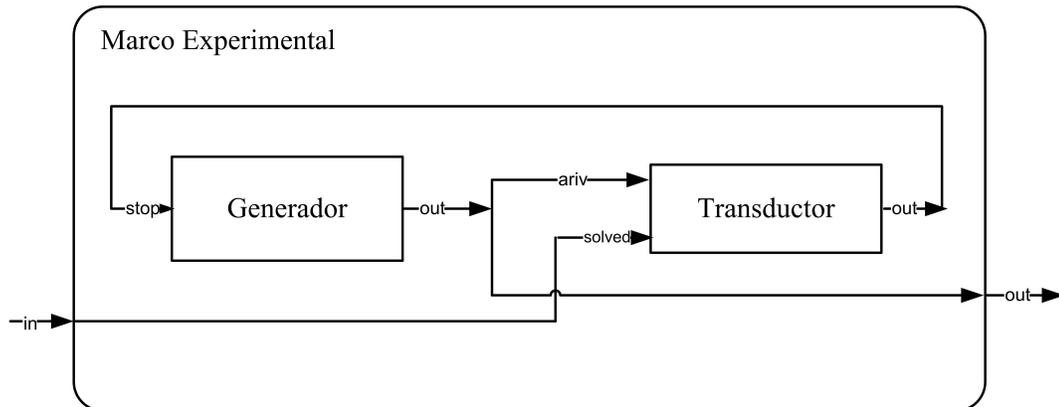


Figura 3.24: Modelo Acoplado del Marco Experimental

A través del puerto *out*, el Generador produce los requerimientos al sistema de

acuerdo a un tiempo entre arribos. Las salidas del Generador se envían al modelo del sistema y al puerto *ariv* del Transductor para que este las registre. Por el puerto *solved* el Transductor recibe los requerimientos atendidos, los que compara con los registrados para calcular las métricas.

Una vez finalizado el tiempo de simulación el Transductor envía por su puerto de salida *out* un mensaje de fin llamado *end* al Generador. Este mensaje es recibido por el puerto de entrada *stop*.

$$\text{Marco Experimental} = \langle X_{me}, Y_{me}, S_{me}, D_{me}, M_i, EIC, IC, EOC \rangle$$

### Conjuntos de puertos de entrada X y salida Y

$$X_{me} = \{in\}$$

$$Y_{me} = \{out\}$$

### Conjunto finito y no vacío de nombres de sub-componentes

$$D_m = \{Generator, Transductor\}$$

### Conjunto de modelos DEVS Atómicos para todo sub-componente $\epsilon D$

$$M_i = \{M_{Generator}, M_{Transductor}\}$$

### Conjunto de acoplamientos de entrada externos (External Input Couplings)

$$EIC = \{((MarcoExperimental, in), (Transductor, solved))\}$$

### Conjunto de acoplamientos internos (Internal Couplings)

$$IC = \{((Generator, out), (Transductor, ariv)), ((Transductor, out), (Generator, stop))\}$$

### Conjunto de acoplamientos de salida externos (External Output Couplings)

$$EOC = \{((Generator, out), (MarcoExperimental, out))\}$$

Por tanto, se establecen acoplamientos con sub-componentes del modelo acoplado desde el exterior (EIC), internos entre sub-componentes (IC) y de salida de sub-componentes hacia el exterior del modelo acoplado (EOC).

### 3.9. Modelo de las Transacciones de Negocios

#### Modelo Atómico Transaccional

El modelo transaccional representa las sesiones de clientes en el sitio de negocios electrónicos. Para construir este modelo se toma como referencia el diagrama CBMG de la Figura 3.25, propuesto en el trabajo de Menascé y otros (2000).

Este patrón representa el comportamiento de clientes frecuentes sobre un sitio que ofrece las transacciones Visitar Página Principal, Navegar, Buscar, Seleccionar productos para ver sus datos, Agregar al Carro de Compras y Pagar. La transición entre cada transacción se efectúa de acuerdo a una probabilidad.

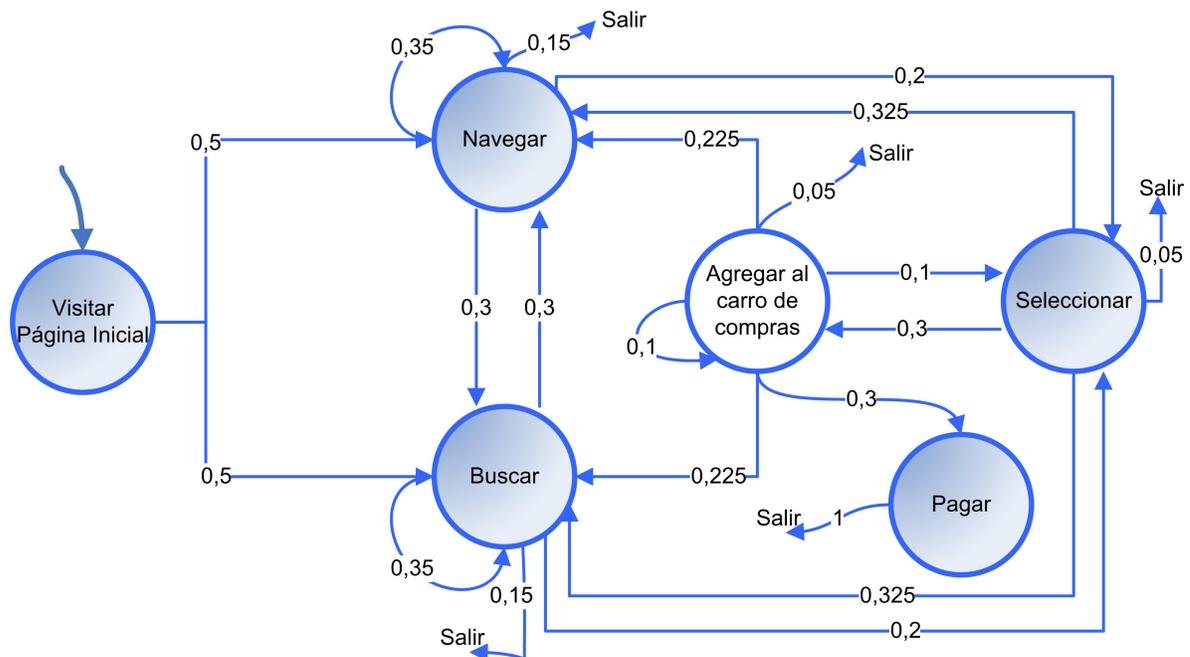


Figura 3.25: CBMG de clientes con comportamiento frecuente de compras

La organización del modelo transaccional comprende dos funcionalidades diferentes: (i) recibir requerimientos de clientes en una transacción y enviarlos a procesar a la plataforma tecnológica y (ii) recibir requerimientos atendidos por la plataforma tecnológica

y enviarlos a la próxima transacción.

En la segunda función del modelo la selección del puerto de salida a la siguiente transacción es aleatoria. Por tanto, la función de salida se transforma en no determinística y de este modo el modelo DEVS es no determinístico. En la Figura 3.26 se especifica el modelo atómico transaccional en un diagrama Statechart.

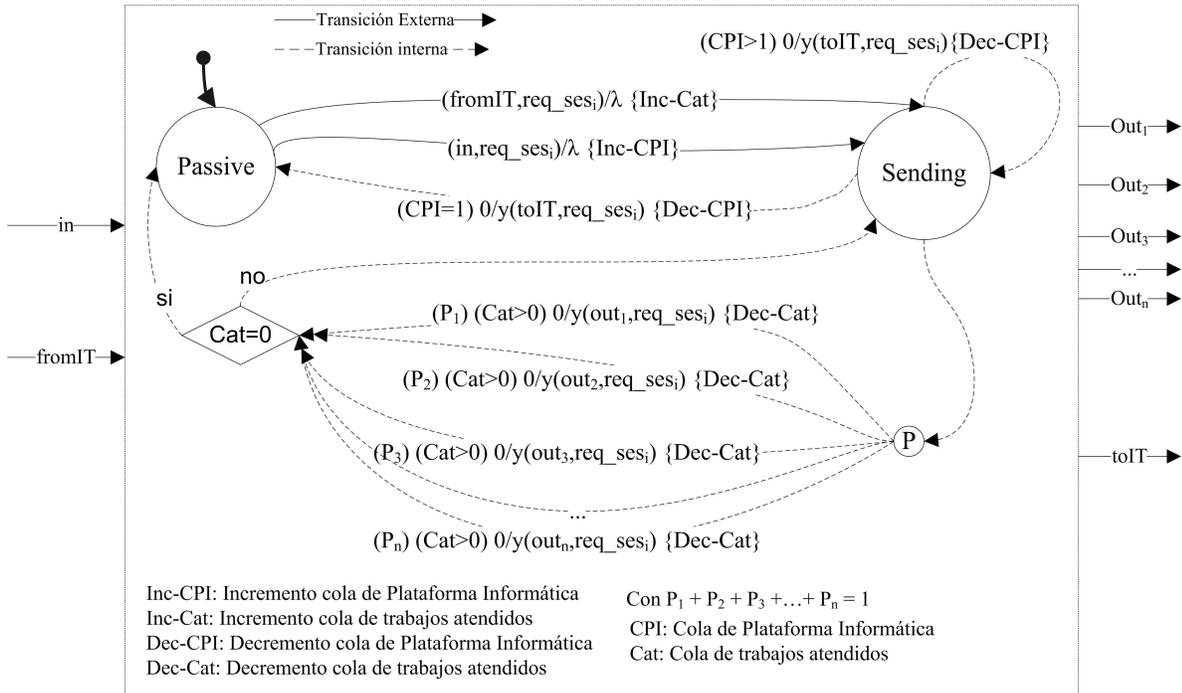


Figura 3.26: Modelo Atómico Transaccional.

Se observa que consta de dos puertos de entradas, uno *in* que recibe los requerimientos a la transacción y otro *fromIT* que recibe los requerimientos atendidos por la tecnología informática. Respecto a los puertos de salida se considera un grupo de *n* puertos *out* cuya selección depende de la probabilidad de transición a la próxima transacción y un puerto *toIT* que envía los requerimientos a ser atendidos desde la tecnología informática. Los estados posibles son *passive* y *sending*, los cuales corresponden a los casos especiales llamados permanente ( $\infty$ ) y transitorio (*tiempo 0*).

En esta representación se incorpora el tratamiento de las colas para enriquecer la

construcción de la lógica del algoritmo. Los requerimientos arribados por el puerto *in* se operan con la Cola de Plataforma Informática, la que se incrementa con una recepción y se decrementa con su procesamiento. Del mismo modo se opera la Cola de Trabajos Atendidos, que se incrementa al recibir un requerimiento atendido por la plataforma informática y se decrementa al ser enviado a la próxima transacción.

El modelo DEVS atómico transaccional no determinístico y paralelo se define como:

$$\text{Modelo Transaccional} = \langle X_{trans}, Y_{trans}, S_{trans}, P_{\lambda, out}, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

### Conjuntos de puertos y valores permitidos

$InputPorts = \{in, fromIT\}$  conjunto de puertos de entrada.

$OutputPorts = \{out_1, out_2, out_3, \dots, out_n, toIT\}$  conjunto de puertos de salida.

$X_{in} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de entrada *in* y representa los requerimientos a una transacción de negocios.

$X_{fromIT} = \{reqses_i\}$  es el conjunto de valores permitidos para el puerto de entrada *fromIT* y representa los requerimientos procesados por la Plataforma Informática.

$Y_{out_i} = \{reqses_i\}$  conjunto de valores permitidos para el puerto de salida *out<sub>i</sub>* y representa el envío de un requerimiento a la próxima transacción de negocios de acuerdo a la probabilidad de transición.

$Y_{toIT} = \{reqses_i\}$  conjunto de valores permitidos para el puerto de salida *toIT* y representa el envío de un requerimiento para su proceso en la Plataforma Informática.

### Conjuntos de eventos de entrada y salida

$X_{trans} = \{(in, reqses_i), (fromIT, reqses_i)\} \mid in, fromIT \in InputPorts, reqses_i \in X_{fromIT}$

$Y_{trans} = \{(out_i, reqses_i), (toIT, reqses_i)\} \mid out_i, toIT \in OutputPorts, reqses_i \in Y_{toIT}$

### Conjunto de estados

$S_{trans} = \{passive, sending\}$

### Conjunto de probabilidades de selección de puerto de salida por transacción

$P_{\lambda,trans_i,out} = \{(p_1, trans_1, out_1), (p_2, trans_2, out_2), \dots, (p_n, trans_n, out_n)\}$  es el conjunto de probabilidades  $p_j$  con  $j = 1..n$ , de pasar de una *transaccin<sub>i</sub>* a las transacciones siguientes por el puerto de salida *out<sub>j</sub>*. La suma de  $p_1 + p_2 + \dots + p_n$  debe ser 1.

### Funciones

Función de transición de estado interna:

$$\delta_{int} : \begin{cases} \delta_{int}(sending) = sending \text{ Si } CPI > 1 \\ \delta_{int}(sending) = passive \text{ Si } CPI = 1 \\ \delta_{int}(sending) = sending \text{ Si } Cat > 1 \\ \delta_{int}(sending) = passive \text{ Si } Cat = 1 \end{cases}$$

Función de transición de estado externa:

$$\delta_{ext} : \begin{cases} \delta_{ext}(passive, e, (in, reqses_i)) = sending \\ \delta_{ext}(passive, e, (fromIT, reqses_i)) = sending \end{cases}$$

Función de transición de confluencia:

$$\delta_{conf}(s, ta(s), x) = \delta_{ext}(\delta_{int}, 0, x)$$

Función de salida estocástica:

$$\lambda(p, out) : \begin{cases} r = \text{random } U[0,1] \\ \lambda(sending) = (out_1, reqses_i) \text{ Si } r \leq P_1 \\ \lambda(sending) = (out_2, reqses_i) \text{ Si } r \leq P_1 + P_2 \\ \lambda(sending) = (out_n, reqses_i) \text{ Si } r \leq P_1 + P_2 + \dots + P_n \end{cases}$$

Función de avance del tiempo:

$$ta : \begin{cases} ta(passive) = \infty \\ ta(sending) = 0 \end{cases}$$

En la función de salida  $\lambda$  se observa que se dividen los intervalos en  $n$  partes, donde cada intervalo tiene asociada una suma de probabilidades. Se genera un número aleatorio  $r$  con una distribución  $U[0, 1]$ , se compara con los diferentes intervalos y de acuerdo al correspondiente se selecciona el puerto de salida.

Frente al arribo de requerimientos concurrentes la función de confluencia es la responsable de organizar su ejecución, de modo que primero habilita la función de transición interna y luego la externa.

### 3.10. Modelo de la Plataforma Tecnológica

En la Sección 3.3 se presenta la teoría de redes de colas y se afirma que los sistemas informáticos pueden ser representados por centros de colas y de demora, lo cual se muestra en la Figura 3.4.

Si es necesario identificar recursos individuales en el sistema se representa la infraestructura como una red de colas, para lo cual se muestran ejemplos como el de las Figuras 3.6, 3.7 y 3.8.

Para el diseño de modelos de infraestructura tecnológica cada recurso individual se identifica con un modelo atómico que representa al centro de cola. La conexión entre recursos se implementa por el acoplamiento de los diferentes modelos atómicos.

Un modelo DEVS atómico de Recurso Tecnológico con cola FIFO se muestra en la Figura 3.27.

El puerto de entrada *in* recibe los requerimientos a ser atendidos y el puerto de salida *out* entrega los requerimientos completados. Los estados posibles son *active* y *passive*. En el estado inicial se encuentra en *passive*.

Su semántica consiste en: dado el arribo de un requerimiento por el puerto *in* la función de transición externa cambia al estado *active* e incrementa la cola de requerimientos. Una vez finalizado el tiempo de servicio se produce una salida de requerimiento atendido por el puerto *out* y el cambio de estado por parte de la función de transición interna está sujeto a los trabajos en cola. De modo que si se encuentra un solo trabajo en cola cambia al estado *passive* y si hay más de uno reinicia el estado en *active*.

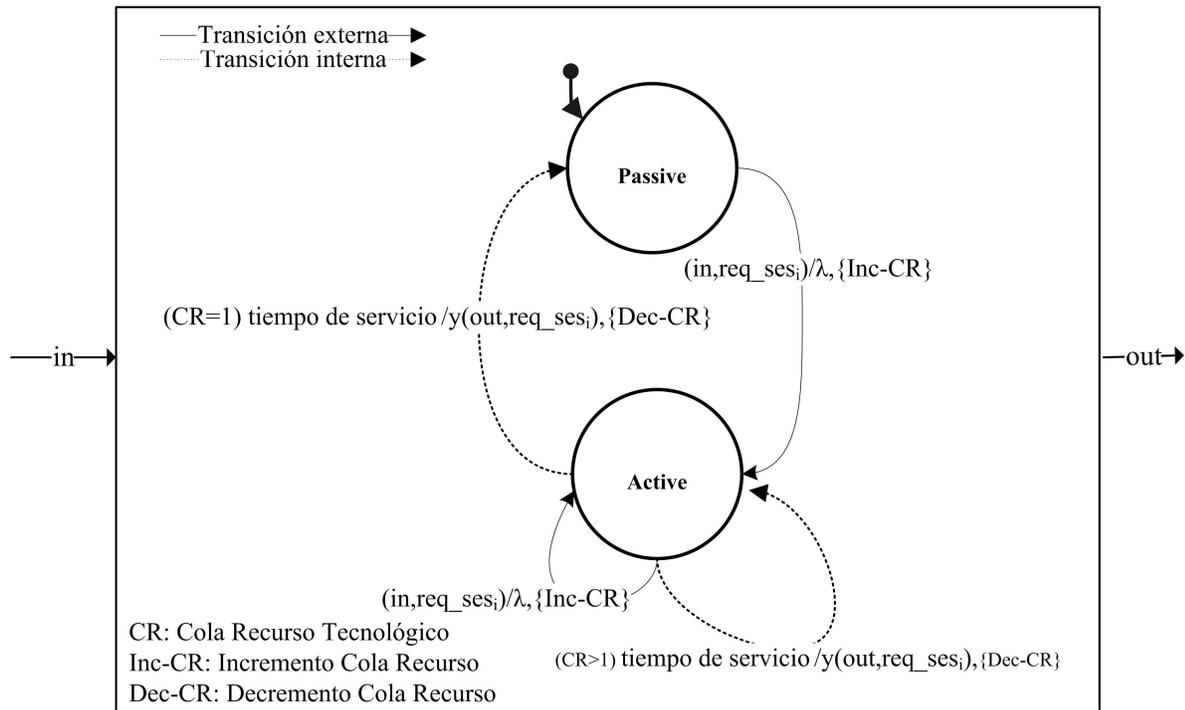


Figura 3.27: Modelo Atómico del Recurso Tecnológico

Si en estado *active* se produce el arribo de un evento externo la función de transición externa almacena en cola el trabajo arribado.

Un modelo DEVS atómico estocástico y paralelo de un recurso físico de hardware denominado Recurso Tecnológico o centro de cola, con disciplina FIFO se representa por la estructura:

$$\text{Recurso Tecnológico} = \langle X_r, Y_r, S, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

### Conjuntos de puertos y valores permitidos

$InputPorts = \{in\}$  conjunto de puertos de entrada.

$OutputPorts = \{out\}$  conjunto de puertos de salida.

$X_{in} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de entrada *in*.

$Y_{out} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de

salida *out*.

### Conjuntos de eventos de entrada y salida

$$X_r = \{(in, reqses_i)\} \mid in \in \text{InputPorts y reqses}_i \in X_{in}.$$

$$Y_r = \{(sal, reqses_i)\} \mid out \in \text{OutputPorts y reqses}_i \in Y_{out}.$$

### Conjunto de estados

$$S = \{active, passive\} \text{ conjunto de estados.}$$

### Funciones

Función de transición de estado interna:

$$\delta_{int} : \begin{cases} \delta_{int}(active) = passive & \text{si CR} = 1 \\ \delta_{int}(active) = active & \text{si CR} > 1 \end{cases}$$

Función de transición de estado externa:

$$\delta_{ext} : \begin{cases} \delta_{ext}(passive, e, (in, reqses_i)) = active \\ \delta_{ext}(active, e, (in, reqses_i)) = active \end{cases}$$

Función de salida:

$$\lambda(active) = Y(out, reqses_i)$$

Función de transición de confluencia:

$$\delta_{conf}(s, ta(s), x) = \delta_{ext}(\delta_{int}, 0, x)$$

Función de avance del tiempo:

$$ta : \begin{cases} ta(active) = \text{tiempo de servicio} \\ ta(passive) = \infty \end{cases}$$

De modo análogo se representa un centro de servicio sin cola, como se muestra en la Figura 3.28, donde todo requerimiento que ingresa es atendido directamente sin necesidad de permanecer en cola y por ello no se necesita la administración de la misma.

A través del acoplamiento de modelos atómicos, ya sea en forma jerárquica o plana, se pueden construir redes de recursos que caractericen diferentes configuraciones de plataformas informáticas.

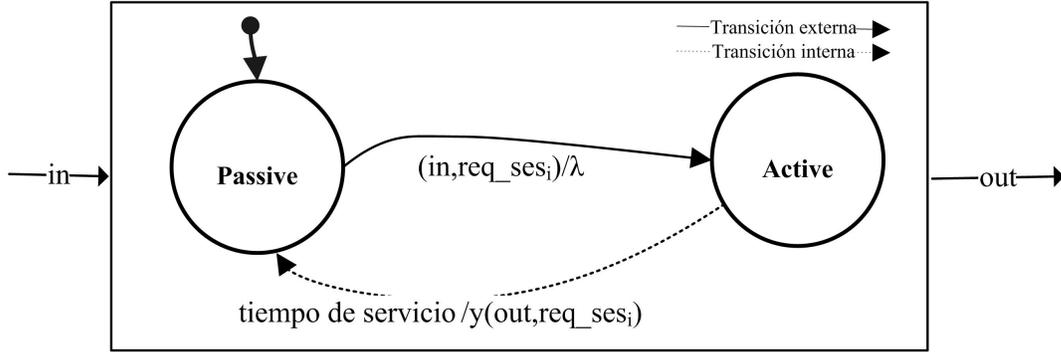


Figura 3.28: Modelo Atómico del Recurso Tecnológico sin Cola

### Modelo DEVS acoplado de Recursos Tecnológicos

Un modelo DEVS acoplado de Recursos Tecnológicos queda definido como una estructura de elementos:

$$MA = \langle X, Y, D, \{M_i \text{ con } i \in D\}, EIC, EOC, IC \rangle$$

$X = \{(p, v)\}; p \in InputPort$  y  $v \in X_p$ , siendo  $InputPort$  el conjunto de puertos de entrada y  $X_p$  los valores de entrada permitidos para el puerto  $p$ .

$Y = \{(p, v)\}; p \in OutputPort$  y  $v \in Y_p$ , siendo  $OutputPort$  el conjunto de puertos de salida y  $Y_p$  los valores de salida para el puerto  $p$ .

$D$ : conjunto de nombres de sub-componentes. Para cada  $i \in D$ :

$M_i = \{X_i, Y_i, S_i, \delta_{int_i}, \delta_{ext_i}, \lambda_i, ta_i\}$  es un modelo DEVS

$EIC \subseteq \{((M, ipN)(d, ipd))\}$  con  $ipN \in InputPort$ ,  $d \in D$  e  $ipd \in InputPort_d$

$EOC \subseteq \{((d, opd)(M, opM))\}$  con  $opM \in OutputPort$ ,  $d \in D$  y  $opd \in OutputPort_d$

$IC \subseteq \{((a, opa)(b, ipb))\}$  con  $a, b \in D$ ,  $opa \in OutputPort_a$ ,  $ipb \in InputPort_b$  y  $a \neq b$

Para describir un acoplamiento se propone un ejemplo de una plataforma tecnológica formada por un router y un servidor. En la Figura 3.29 se presenta el modelo de red de colas.

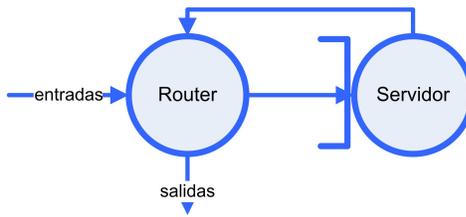


Figura 3.29: Modelo de Red de Colas

El router se representa por un centro de demora y el servidor por un centro de cola. Los requerimientos que arriban son atendidos por el router, el cual los envía al servidor. Si el servidor está desocupado los atiende, en caso contrario permanecen en cola hasta que éste se desocupe. Una vez procesados, los requerimientos son devueltos al router para su posterior salida.

A partir del modelo de red de colas se elabora el diagrama Statechart. En la Figura 3.30 se presenta el modelo acoplado Plataforma Informática compuesto por los modelos atómicos router y servidor Web.

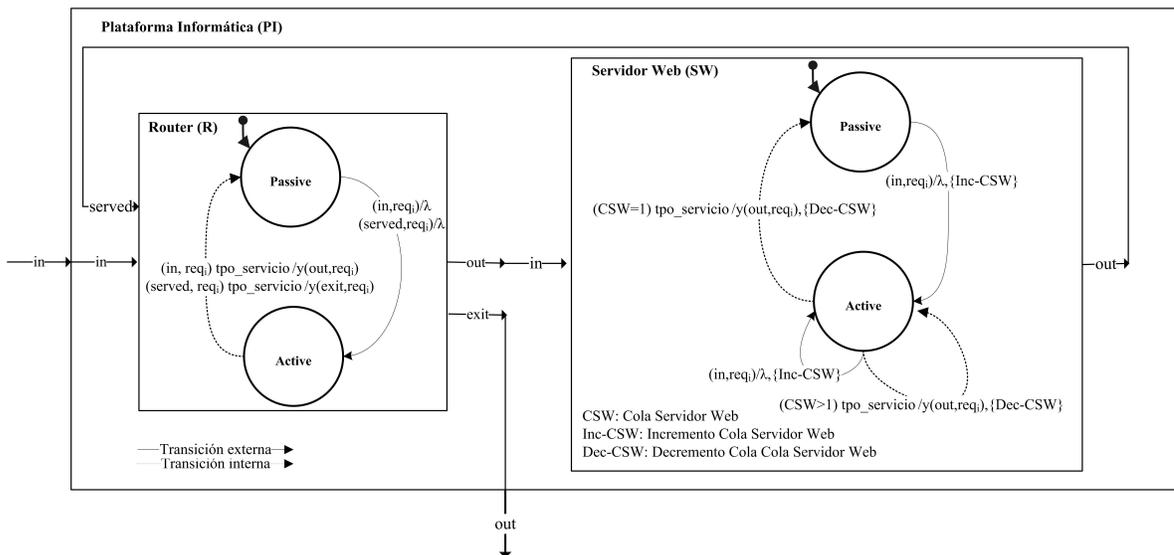


Figura 3.30: Modelo Acoplado de Plataforma Informática

El modelo atómico router no administra cola y contiene los puertos de entrada *served* para recibir los requerimientos atendidos por el servidor Web y otro *in* para

ingresar los requerimientos que arriban a la plataforma informática. Por el puerto de salida *out* envía el requerimiento al servidor y por el *exit* entrega los requerimientos atendidos. El modelo atómico servidor Web administra una cola, recibe los requerimientos del router por el puerto de entrada *in* y los devuelve por el puerto de salida *out*.

En el modelo de Plataforma Informática se observan los siguientes acoplamientos: un acoplamiento de entrada externo de los puertos *in* de plataforma informática con *in* del router, un acoplamiento de salida externo *exit* del router con *out* de la plataforma informática y dos acoplamientos internos (i) *out* del router con *in* del servidor Web y (ii) *out* del servidor Web con *served* del router.

Un modelo DEVS acoplado de Plataforma Informática queda definido como una estructura de elementos:

$$\text{Plataforma Informática} = \langle X_{PI}, Y_{PI}, D, M_i, EIC, IC, EOC \rangle$$

$$X_{PI} = \{in\}$$

$$Y_{PI} = \{out\}$$

$$D = \{Router, ServidorWeb\}$$

$$M_i = \{M_r, M_{sw}\}$$

$$EIC = \{((PI, in), (R, in))\}$$

$$IC = \{((R, out), (SW, in)), ((out, SW), (served, R))\}$$

$$EOC = \{((R, exit), (PI, out))\}$$

### 3.11. Conclusiones

En este capítulo se presentan técnicas de modelado de sistemas de evento discreto y se muestra su aplicación en comercio electrónico. Se parte de la definición de diagramas de transición de estados para mostrar su implementación en diferentes técnicas.

Primero se presenta el gráfico CBMG como herramienta de modelado de sesiones de clientes, el cual consiste en un conjunto de estados y arcos dirigidos entre estados. Cada estado se corresponde con una transacción que un cliente puede ejecutar en el proceso de comercio electrónico y cada arco dirigido con una transición entre transacciones. Las transiciones están sujetas a una probabilidad de ocurrencia. De este modo se pueden representar los diferentes caminos que un cliente puede tomar desde su ingreso al sitio de negocios hasta el abandono del mismo.

Del tratamiento analítico del CBMG se pueden obtener medidas tales como el número de visitas a cada transacción y así encontrar un mapa de cargas a los recursos que ejecutan las transacciones.

La teoría de redes de colas propone estrategias para evaluar el desempeño de un sistema informático como una red de centros de cola o demora. Cada recurso informático es representado como un centro de cola, si el mismo tiene la capacidad de mantener colas de espera, en caso contrario es un centro de demora. Aplicando las leyes fundamentales se obtienen métricas de desempeño del sistema informático.

Las técnicas hasta aquí expuestas no permiten la identificación de estados. Por ello se analiza la posibilidad de trabajar con redes de Petri y diagramas Statechart.

Las redes de Petri se construyen en función de los estados posibles del proceso de comercio electrónico, transiciones que indican el cambio de estado frente al arribo de un evento y tokens que materializan eventos.

El diagrama Statechart es también una herramienta reactiva y se elabora en función de los estados del sistema y las transiciones entre estados. Es importante destacar la posibilidad de especificar la acción correspondiente a cada transición y la incorporación de condiciones. De este modo dicha acción no está sujeta solo a la ocurrencia de un evento sino opcionalmente a la evaluación de una condición.

Mediante un modelo de servidor Web, se muestra el diseño en redes de colas, redes de Petri y Statechart. De la comparación de las tres técnicas se concluye:

- **Teoría de Colas:** simplicidad en la descripción de los procesos de colas de un sistema, no se muestran los eventos que determinan la dinámica del sistema, incapacidad para identificar estados.
- **Redes de Petri:** permite la identificación de estados y con la utilización de tokens de color propone extensiones a modelos jerárquicos y estocásticos; se torna ininteligible para modelos complejos.
- **Statechart:** además de la identificación de estados, representa claramente el paralelismo del modelo, con la identificación de acciones en las transiciones ayuda a describir las funciones de transición, muestra elementos que ayudan a delinear la semántica, posibilitan una construcción jerárquica y agregan componentes estocásticos.

Una alternativa para la obtención de métricas de desempeño de procesos de comercio electrónico es la simulación. Para ello es necesario la construcción de un programa que reproduce el comportamiento del sistema de acuerdo al modelo propuesto.

Una técnica de especificación de modelos para simulación de sistema de evento discreto es DEVS. Mediante el cual se cuenta con un formalismo que propone estrategias

para la construcción rigurosa de modelos con capacidad de representación de sistemas complejos.

Además, con la extensión a DEVS paralelo se incorporan capacidades de ortogonalidad y con DEVS estocástico se agregan bondades para la aleatorización del comportamiento del sistema.

Se proponen las principales entidades del framework DEVS para modelización de transacciones electrónicas. Estas entidades se encuentran agrupadas en dos estructuras fundamentales: el Marco Experimental y el Modelo de Sistema. El Marco Experimental contiene los modelos atómicos Generador y Transductor. El Modelo de Sistema describe al proceso de comercio electrónico y está compuesto por dos modelos acoplados, el Transaccional y la Plataforma Informática.

A nivel transaccional se diseña el proceso de negocios, mientras que en la plataforma informática se construye el soporte tecnológico que implementa el proceso mencionado.

Para cada uno de los componentes acoplados y atómicos se presentan los diagramas Statechart, a partir de los cuales se elabora la semántica del formalismo DEVS. De la observación de los diagramas Statechart se determinan los puertos de entrada y salida, los estados y sus funciones de transición interna y externa.

De este modo se propone una técnica de construcción del formalismo DEVS que orienta el diseño de los algoritmos de forma clara y simple.



# CAPÍTULO 4

## Construcción y Simulación de Modelos de Comercio Electrónico

*En este capítulo se muestra la aplicación de la metodología de modelización y simulación propuesta mediante la elaboración de un modelo de proceso de comercio electrónico tipo B2C. La construcción del modelo se basa en un gráfico CBMG de transacciones electrónicas tomado de un caso de estudio bibliográfico, mientras que la plataforma informática se compone de una arquitectura tecnológica basada en un cluster de servidores. La coreografía del modelo se apoya en la integración del sitio de negocios con un servicio de cobro electrónico. Se desarrollan las diferentes etapas y se simula el comportamiento para determinadas frecuencias de arribos de clientes. Se presentan los resultados de simulación y se emplea el diseño experimental estadístico como la estrategia de análisis de salidas.*

## 4.1. Descripción del Problema

El modelo de comercio electrónico a construir se elabora en base al trabajo de los autores Menascé y otros (2000). Del mismo se toma el problema de negocios consistente en la evaluación del desempeño de un proceso de comercio electrónico para la venta minorista de libros y se consideran algunos aspectos tales como el diagrama CBMG y las cargas de clientes al sistema.

Para el modelo de proceso de comercio electrónico se propone una plataforma informática orientada a servicios con tecnologías de servicios Web.

## 4.2. Construcción del Modelo de Negocios

Para la aplicación de la metodología propuesta en el Capítulo 2 se requiere el diseño de un proceso de comercio electrónico a través de la utilización de un sitio de negocios para la venta minorista de libros. Las operaciones de negocios se implementan en un conjunto de transacciones que representan sesiones de clientes que navegan por el sitio, buscan datos de un producto, agregan al carro de compras y pagan.

La interacción entre transacciones obedece al comportamiento estocástico de los clientes en el recorrido del sitio y se implementan con tecnologías orientadas a servicios.

El pago se realiza en línea con tarjeta de crédito y se ejecuta como un servicio tercerizado.

Por tanto, la estrategia de negocios consiste en la venta minorista en línea de libros, con implementación de operaciones totalmente virtuales, teniendo como objetivo general la elaboración de un modelo de simulación DEVS de un proceso de comercio electrónico orientado a servicios y la evaluación del desempeño tecnológico en relación con los beneficios obtenidos.

Para lograr el objetivo general se plantean los siguientes objetivos específicos:

- Construir el modelo de composición de servicios del modelo de negocios.
- Monitorizar el uso de servicios Web y obtener los parámetros del tiempo de servicio de cada transacción electrónica.
- Configurar la plataforma informática que soporte el proceso de comercio electrónico.
- Simular el sistema de comercio electrónico en relación al comportamiento tecnológico, obtener métricas integradas y presentar salidas.

### **Modelo de Negocios**

La empresa debe diseñar un sistema de comercio electrónico eficiente que ofrezca al cliente navegabilidad con indicadores de tiempo de respuesta adecuados y a su vez muestre confiabilidad para el cobro por medios electrónicos.

Para la ejecución del cobro es conveniente la contratación de un servicio externo que sea eficiente y además que genere confianza en los clientes. Paypal es una empresa dedicada al cobro de productos o servicios comercializados por medios electrónicos. Por cada transacción, percibe una comisión entre el 3 % al 4 %, según el contrato de servicio.

Para la comunicación entre el sitio de comercio electrónico ofrece aplicaciones implementadas con tecnologías de servicios Web, publicando las aplicaciones como servicios a ser invocadas por el cliente por mensajes SOAP.

Las características del negocio estiman un alto consumo de recursos informáticos, requieren disponibilidad continua del sitio y escalabilidad en horarios picos o fechas comerciales y el análisis de salida esperado debe integrar aspectos tecnológicos y de negocios.

En el modelo se proponen dos participantes, el propietario de sitio y la empresa de cobro electrónico a través de un servicio tercerizado. El propietario del sitio tiene la función de implementar el sistema de comercio electrónico y la gestión del sitio, mientras que Paypal es la empresa responsable del cobro electrónico.

En cuanto a las transacciones electrónicas, el propietario del sitio es responsable de la administración de la página inicial del sitio, su navegación, consulta de información sobre un producto y gestión del carro de compras, mientras que solo el cobro electrónico está a cargo de Paypal.

Las transacciones se identifican como:

- **Visit Homepage:** el usuario ingresa al sitio de comercio electrónico por la página principal.
- **Browse:** navegación a través de las páginas.
- **Search:** búsqueda de datos de un libro en particular.
- **Select Product:** selección del libro de interés.
- **Add Cart:** agregado del libro seleccionado al carro de compras.
- **Pay:** pago electrónico de la orden de compra.

En la Tabla 4.1 se presentan los responsables de la implementación de las diferentes transacciones del sitio de comercio electrónico.

### 4.3. Elaboración del Proceso de Comercio Electrónico

La elaboración del proceso de comercio electrónico se fundamenta en herramientas de composición de servicios y se describen como coreografía y orquestación.

Transacciones	Responsable
Visit Homepage	Propietario del sitio
Browse	Propietario del sitio
Search	Propietario del sitio
Select Product	Propietario del sitio
Add Cart	Propietario del sitio
Pay	Paypal

Tabla 4.1: Transacciones y responsables de las mismas

### 4.3.1. Coreografía de Interacciones con Servicio Tercerizado

En la coreografía se modela la interacción entre el propietario del sitio y Paypal, lo que se detalla en el diagrama de secuencia de la Figura 4.1.

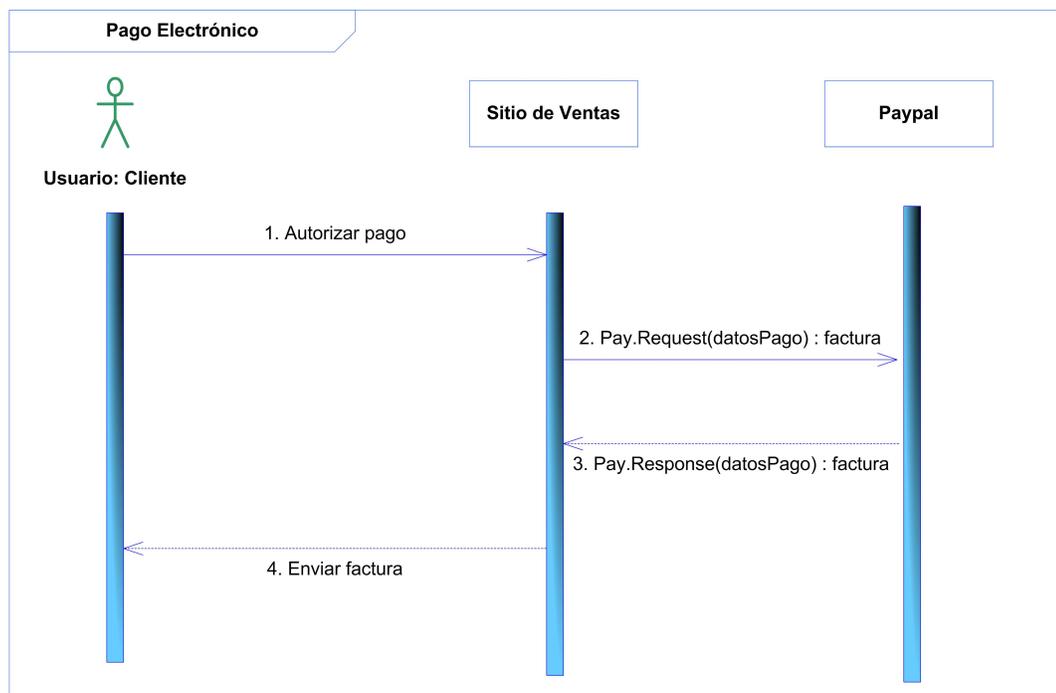


Figura 4.1: Diagrama de secuencia de integración con servicio de cobro

Ante la autorización del pago por parte del cliente, el sistema del sitio de ventas envía un mensaje solicitando el cobro a la empresa Paypal.

Paypal tiene activo el proveedor de servicio Web para el cobro electrónico, por lo

que al recibir el mensaje *Pay.Request* ejecuta la operación y devuelve en el mensaje *Pay.Response* la constancia correspondiente.

### 4.3.2. Orquestación de las Transacciones Internas

Se toma el gráfico CBMG de la Figura 4.2 (Menascé y otros, 2000) para representar las transacciones del sitio de comercio electrónico y sus probabilidades de transición. El diagrama seleccionado comprende a compradores habituales, con un porcentaje mayor de probabilidad de transición de la transacción *Add Cart* a *Pay*.

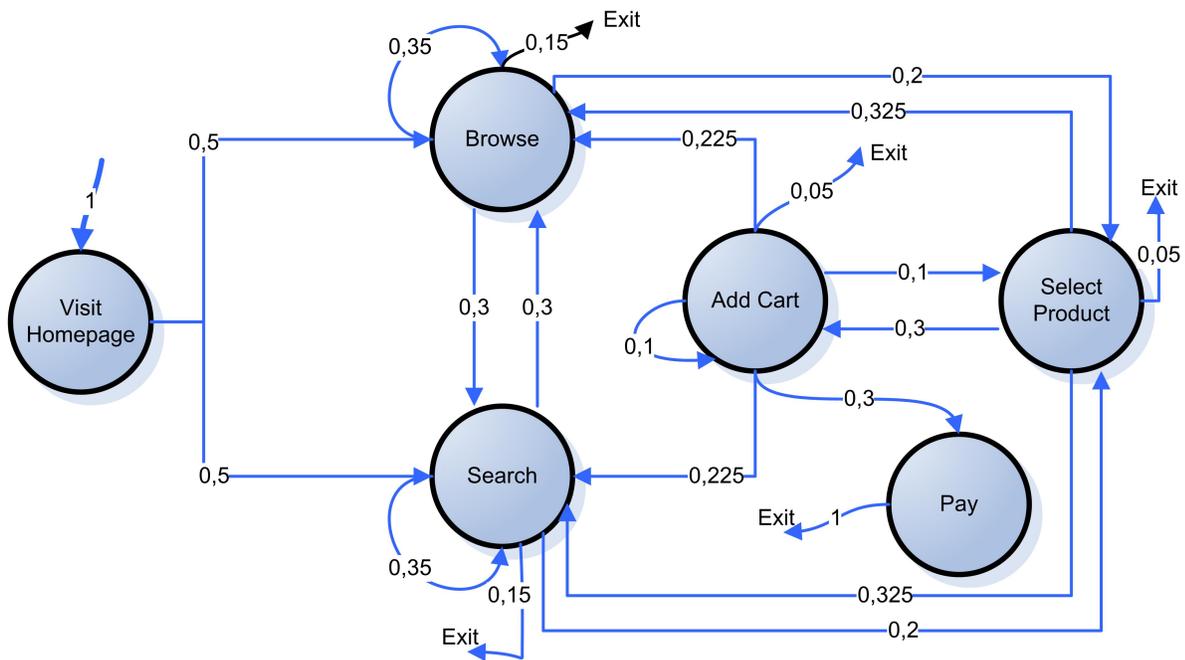


Figura 4.2: Customer Behavior Model Graph de compradores frecuentes

La sesión comienza por la transacción *Visit Homepage*, de la cual se puede continuar navegando en el sitio (*Browse*) o realizar búsquedas de datos de productos (*Search*). Si el producto encontrado es de interés se lo selecciona (*Select Product*) como paso previo a su incorporación en el carro de compras (*Add Cart*). Una vez decidida la compra, el cliente puede autorizar el pago desde el carro de compras (*Pay*).

La transacción *Exit* se indica en la flecha de la transición, no se grafica en un círculo para claridad de la representación.

Los diferentes caminos a seguir por el cliente se indican por las transiciones entre transacciones. Por ejemplo, si el cliente está en la transacción *Search* tiene 20 % de probabilidad de pasar a *Select Product*, 30 % de ir a *Search*, 35 % de volver a *Search* y 15 % de abandonar el sitio (*Exit*).

#### 4.4. Planteo del Modelo de Recursos

La tecnología informática a utilizar debe asegurar disponibilidad del sistema de comercio electrónico y escalabilidad frente a picos de requerimientos de clientes.

Por ello, se implementa un cluster de  $n$  servidores paralelos cuya configuración hace posible que cada uno de los recursos pueda responder a las diferentes transacciones del sistema y de este modo balancear las cargas de trabajo.

Si algún servidor dejara de funcionar, existen  $n-1$  servidores capaces de realizar las mismas tareas.

El modelo de red de colas del cluster se presenta en la Figura 4.3. Consiste en un centro de cola que representa al balanceador de cargas y en  $n$  centros de demora por cada servidor. El balanceador de carga gestiona la recepción de los requerimientos arribados con una asignación secuencial en los servidores y es responsable de mantener en cola los arribos que no pueden ser atendidos. Los servidores no administran colas por ello se modelan como centros de demora.

Respecto a las fallas de los recursos y de acuerdo a experiencias previas en servidores, se pueden distinguir dos tipos principales:

- Rotura de componente: se produce a intervalos anuales y el tiempo para resolverlo es el demorado en reemplazar el equipo (uno al año con tres a diez días para

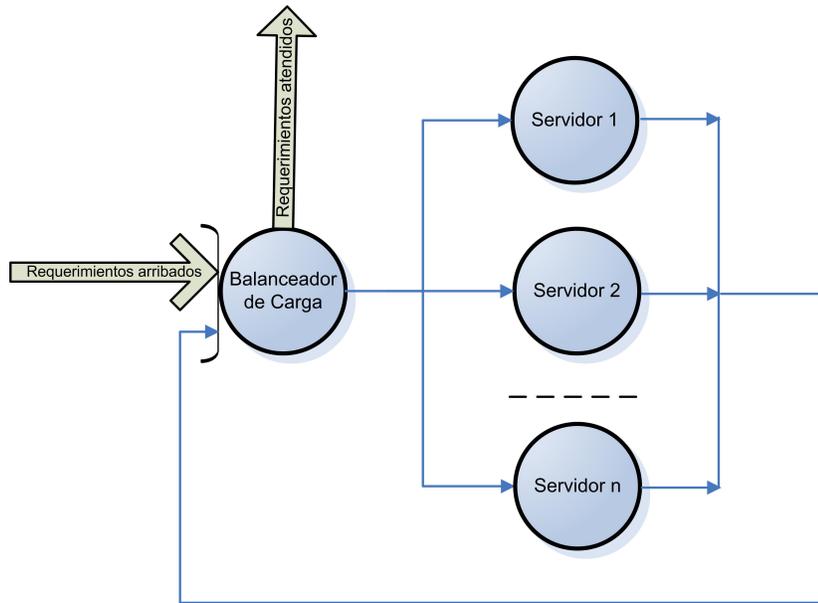


Figura 4.3: Modelo de red de colas del cluster de servidores

resolverlo).

- Calidad del enlace: se produce 240 horas por año aproximadamente. Se estima que no habrá conectividad diez días del año en total y se resuelve contratando a más de un proveedor de Internet, con un procedimiento automático que recupera el servicio a través de la comunicación por otro enlace.

Por ello presenta el caso con la consideración de fallas en los servidores, con posibilidad de recuperación del hardware y de la transacción en ejecución.

Desde el punto de vista de la tecnología de software, las transacciones se implementan como operaciones con servicios Web.

## 4.5. Esquematización de Métricas de Desempeño

Desde el enfoque tecnológico que tiene en cuenta el desempeño de los recursos de la plataforma informática, se consideran las siguientes métricas:

- Tiempo de Respuesta [seg.]
- Velocidad de Procesamiento [req./seg.].
- Cantidad de requerimientos en cola [req.].

Desde el enfoque de negocios se consideran las siguientes métricas:

- Cantidad de productos vendidos [productos/día].
- Ingresos por ventas [U\$D].
- Ganancias Netas [U\$D].
- Punto de Equilibrio [U\$D]

## 4.6. Construcción del Modelo DEVS

La técnica de elaboración del modelo DEVS comienza con la construcción de un diagrama de estructura que identifique los principales componentes del modelo y sus relaciones y luego se bosqueja un modelo base con los componentes atómicos y acoplados.

### 4.6.1. Diagrama de Estructura de las Entidades del Sistema

La construcción del modelo de simulación comienza con la identificación de las entidades y sus relaciones. En la Figura 4.4 se muestra el diagrama de estructura del proceso de comercio electrónico para simulación.

El modelo de simulación está compuesto por el Marco Experimental y el Modelo de Comercio Electrónico. Como ya se explicó en la Sección 3.8, el Marco Experimental

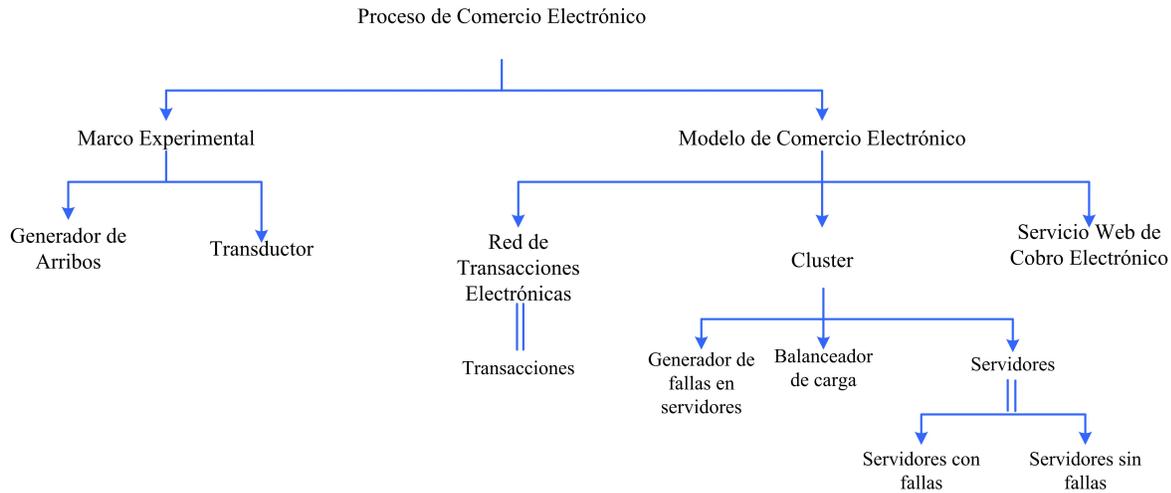


Figura 4.4: Diagrama de estructura del modelo de comercio electrónico

contiene a un Generador de las cargas de clientes y un Transductor que recolecta datos para computar las métricas.

El Modelo de Comercio Electrónico contiene por un lado el nivel transaccional definido por una red de transacciones y por otro, el tecnológico, compuesto por el Cluster y el servicio Web para el cobro electrónico.

El Cluster consiste en un Balanceador de Carga, Servidores que lo componen y un Generador de Fallas a los servidores. Por ello de acuerdo a las fallas generadas, los servidores pueden estar en fallas o en estado de funcionamiento correcto.

### 4.6.2. Modelo Base

En la Figura 4.5 se esquematiza el modelo base, compuesto por los submodelos Experimental Framework, Transactional Model y Technological Model.

El Experimental Framework es un modelo acoplado que contiene el Generator y el Transducer. El Generator produce los arribos de requerimientos por el puerto *out* y recibe por el *stop* el mensaje de fin de simulación. El Transducer recibe por el puerto

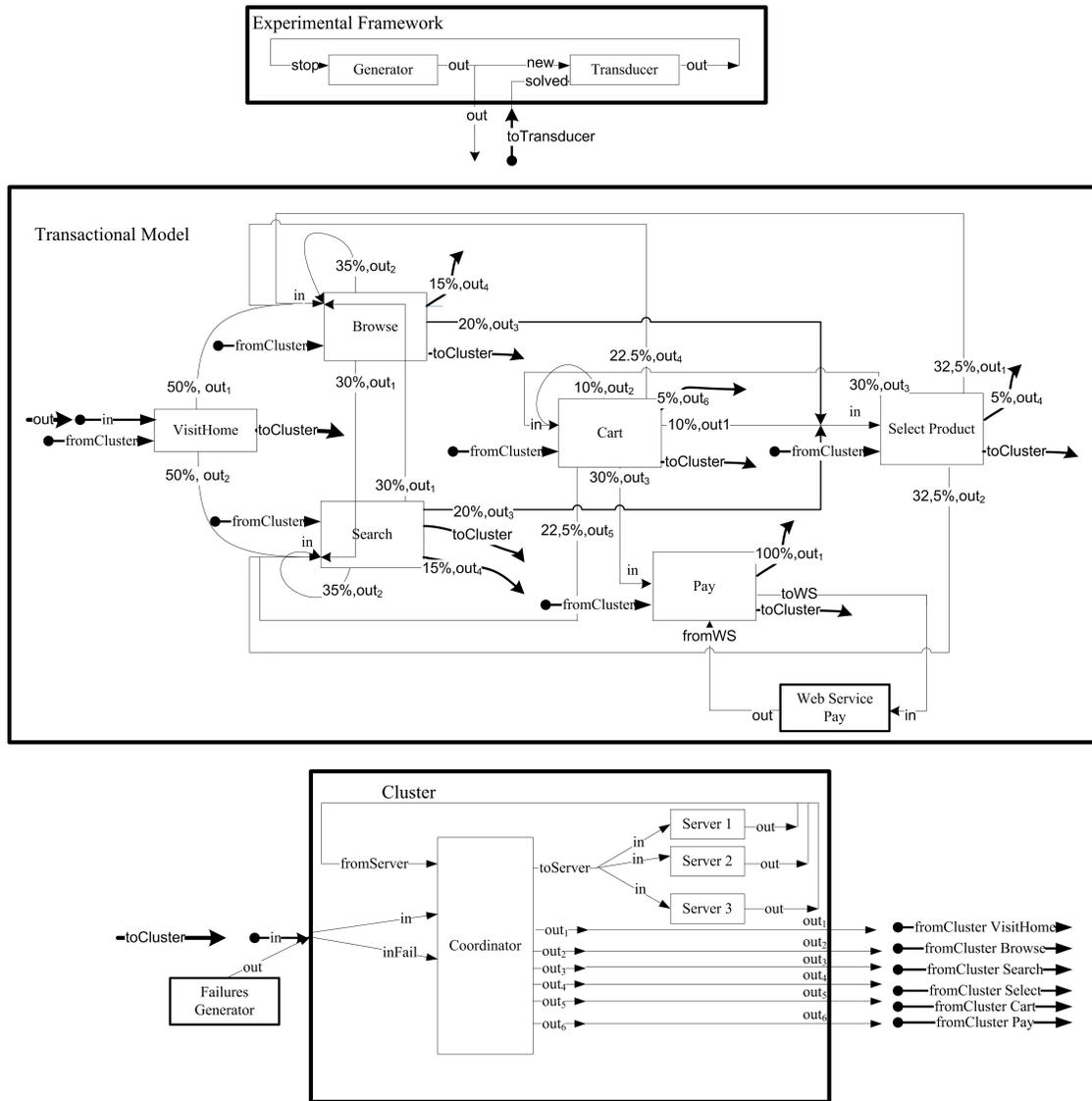


Figura 4.5: Modelo base de componentes

*new* cada requerimiento generado, por el *solved* los atendidos y por el puerto *out* envía el mensaje de fin de simulación.

Cada requerimiento generado inicia una nueva sesión de cliente y es enviado al Transactional Model por el puerto *out*. El mensaje de una sesión que finaliza se recibe por el puerto *toTransducer*.

El Transactional Model es un modelo acoplado compuesto por las interacciones de

las transacciones del sitio modelado en el diagrama CBMG. Un requerimiento que arriba a una transacción pasa a ser atendido por la plataforma informática de acuerdo a un tiempo de servicio, una vez que es procesado continúa con la siguiente transacción. La transición entre transacciones es aleatoria de acuerdo a una probabilidad.

Cada transacción es un modelo atómico y se distinguen:

- **De inicio de sesión:** en la transacción *VisitHome* el inicio de una sesión se produce por el puerto *in*, los requerimientos atendidos son recibidos por el puerto *fromCluster*, los requerimientos a procesar son enviados por el puerto de salida *toCluster* y a la próxima transacción por el puerto de salida *out<sub>i</sub>*.
- **Intermedias:** los puertos de entrada son *in* para recibir un requerimiento desde la transacción anterior y *fromcluster* para los requerimientos que fueron atendidos por la plataforma informática, por el puerto de salida *toCluster* se envían a procesar y a la próxima transacción por el puerto de salida *out<sub>i</sub>*. Además cuentan con un puerto de salida *out* que envía el fin de sesión al Transducer. Son el caso de *Browse*, *Search*, *Cart* y *Select Product*.
- **Pay:** contiene la misma descripción que la intermedia con el adicional de un puerto de salida *toWS* por el cual se envían los requerimiento al Web Service Pay.
- **Web Servcie Pay:** recibe un requerimiento por el puerto de entrada *in*, lo procesa y lo devuelve por el puerto de salida *out*.

Tecnological Model es un modelo acoplado que implementa la plataforma informática como un cluster de servidores paralelos. Recibe por el puerto *in* requerimientos desde cada transacción, los que son atendidos por el coordinador (Coordinator), quien los pone en cola y los asigna en forma secuencial al siguiente servidor libre a través de un puerto de salida *toServer*.

Coordinator tiene la capacidad de distinguir si los requerimientos son los arribados o ya fueron atendidos por un servidor. Para un arribo, el servidor atiende el requerimiento por el puerto de entrada *in* y lo devuelve procesado por el puerto de salida *out*; en caso contrario son devueltos a la transacción por un puerto *out<sub>i</sub>*.

Además la plataforma informática puede fallar. Por ello se considera un generador de fallas llamado Failures Generator que produce un indicador de fallas. Estas fallas son recibidas por el Coordinator a través del puerto *inFail*, el cual reacciona y la aplica en un servidor.

Cada Server tiene la capacidad de recuperarse luego de un tiempo determinado y el Coordinator de reiniciar el requerimiento no completado.

## 4.7. Especificación del Modelo de Comercio Electrónico en el Formalismo DEVS

Tomando como referencia el modelo base de la Figura 4.5, se elaboran los diagramas Statechart de los modelos atómicos y acoplados.

Dichos modelos son base para la especificación del formalismo DEVS, a través de la identificación de variables de estado, puertos de entrada y salida, funciones de transición de estados y de avance del tiempo.

### 4.7.1. Modelo del Marco Experimental

En la Figura 4.6 se muestra el diagrama Statechart del Experimental Framework, la cual consta de la descripción en paralelo de los modelos Generator y Transducer.

El modelo atómico Generator produce los arribos de requerimientos con un tiempo entre arribos determinado por una distribución de probabilidad. La generación de

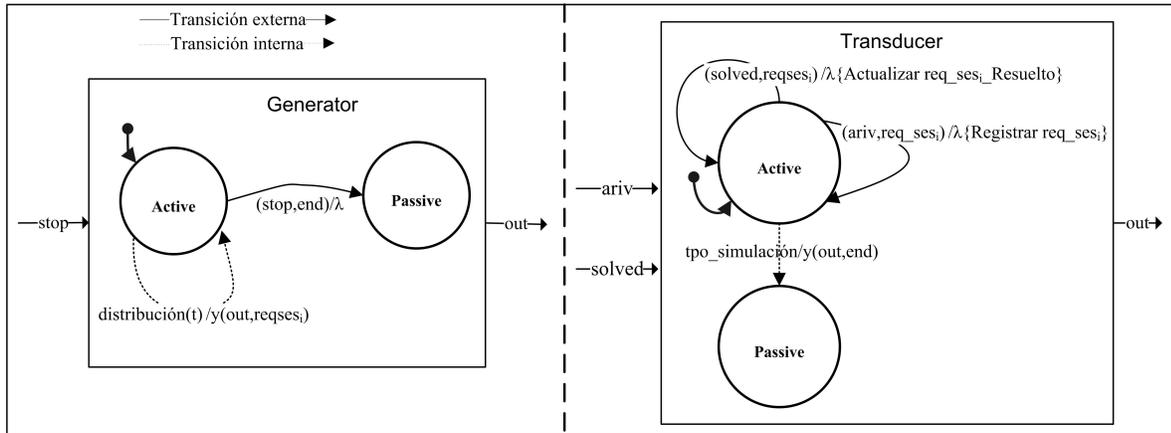


Figura 4.6: Diagrama Statechart del Modelo Acoplado del Experimental Framework

arribos se realiza hasta recibir por el puerto *stop* un mensaje de fin de simulación.

El modelo atómico Transducer recibe del Generator la información de los requerimientos iniciados a través del puerto *ariv* y espera la finalización de los mismos por el puerto *solved*. Con la información de requerimientos arribados, resueltos y el tiempo de simulación se calculan las métricas del tiempo de respuesta y la velocidad de procesamiento. Una vez finalizado el tiempo de simulación, envía un mensaje al Generator por el puerto *out*.

La especificación en el formalismo DEVS del Experimental Framework se desarrolló en la Sección 3.8. De modo que a partir del diagrama se construye el formalismo. También en la misma sección se detalla el acoplamiento interno entre los dos modelos atómicos Generator y Transducer, así como el externo con el modelo acoplado Transaccional.

### 4.7.2. Modelo Transaccional

#### Modelo Atómico de Transacciones

En la Figura 3.26 de la la Sección 3.9 se propuso el diagrama Statechart del modelo

atómico de transacciones, cuya semántica consiste en (i) recibir los requerimientos por el puerto *in*, que pueden provenir del Experimental Framework como sesiones de clientes que se inician o de otra transacción, (ii) enviar los requerimientos recibidos por el puerto *in* a procesar a la plataforma tecnológica (*tocluster*); (iii) recibir requerimientos procesados por el puerto *fromcluster* y (iv) decidir aleatoriamente el puerto de salida  $out_1, out_2, \dots, out_n$  a la próxima transacción.

Como complemento a lo presentado en la Sección 3.9 se definen los modelos atómicos específicos, con la descripción de los puertos de entrada y salida, conjunto de probabilidades y la función estocástica de salida.

**Transacción Visit Homepage:**

$$\text{VisitHomepage} = \langle X_{VH}, Y_{VH}, S_{VH}, P_{VH}, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

*InputPorts* = {*in*, *fromCluster*} conjunto de puertos de entrada.

*OutputPorts* = {*out<sub>1</sub>*, *out<sub>2</sub>*, *toCluster*} conjunto de puertos de salida.

$$P_{\text{VisitHomepage}} = \{(50\%, \text{Browse}, out_1), (50\%, \text{Search}, out_2)\}$$

$$\lambda : \begin{cases} r = \text{random U}[0,1] \\ \lambda(\text{sending}) = (out_1, reqses_i) \text{ Si } r \leq 0,5 \\ \lambda(\text{sending}) = (out_2, reqses_i) \text{ Si } r \leq 1 \end{cases}$$

**Transacción Browse:**

$$\text{Browse} = \langle X_B, Y_B, S_B, P_B, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

*InputPorts* = {*in*, *fromCluster*} conjunto de puertos de entrada.

*OutputPorts* = {*out<sub>1</sub>*, *out<sub>2</sub>*, *out<sub>3</sub>*, *out<sub>4</sub>*, *toCluster*} conjunto de puertos de salida.

$$P_{\text{Browse}} = \{(30\%, \text{Search}, out_1), (35\%, \text{Browse}, out_2), (20\%, \text{Select}, out_3), (15\%, \text{End}, out_4)\}$$

$$\lambda : \begin{cases} r = \text{random } U[0,1] \\ \lambda(\text{sending}) = (out_1, reqses_i) \text{ Si } r \leq 0,3 \\ \lambda(\text{sending}) = (out_2, reqses_i) \text{ Si } r \leq 0,65 \\ \lambda(\text{sending}) = (out_3, reqses_i) \text{ Si } r \leq 0,85 \\ \lambda(\text{sending}) = (out_4, reqses_i) \text{ Si } r \leq 1 \end{cases}$$

**Transacción Search:**

$$\text{Search} = \langle X_S, Y_S, S_S, P_S, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

*InputPorts* = {*in, fromCluster*} conjunto de puertos de entrada.

*OutputPorts* = {*out<sub>1</sub>, out<sub>2</sub>, out<sub>3</sub>, out<sub>4</sub>, toCluster*} conjunto de puertos de salida.

$$P_{Search} = \{(30\%, Browse, out_1), (35\%, Search, out_2), (20\%, Select, out_3), (15\%, End, out_4)\}$$

$$\lambda : \begin{cases} r = \text{random } U[0,1] \\ \lambda(\text{sending}) = (out_1, reqses_i) \text{ Si } r \leq 0,3 \\ \lambda(\text{sending}) = (out_2, reqses_i) \text{ Si } r \leq 0,65 \\ \lambda(\text{sending}) = (out_3, reqses_i) \text{ Si } r \leq 0,85 \\ \lambda(\text{sending}) = (out_4, reqses_i) \text{ Si } r \leq 1 \end{cases}$$

**Transacción Cart:**

$$\text{Cart} = \langle X_C, Y_C, S_C, P_C, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

*InputPorts* = {*in, fromCluster*} conjunto de puertos de entrada.

*OutputPorts* = {*out<sub>1</sub>, out<sub>2</sub>, out<sub>3</sub>, out<sub>4</sub>, out<sub>5</sub>, out<sub>6</sub>, toCluster*} conjunto de puertos de salida.

$$P_{Cart} = \{(10\%, Select, out_1), (10\%, Cart, out_2), (30\%, Pay, out_3),$$

$$(22,5\%, Browse, out_4), (22,5\%, Search, out_5), (5\%, End, out_6)\}$$

$$\lambda : \left\{ \begin{array}{l} r = \text{random U}[0,1] \\ \lambda(\text{sending}) = (out_1, reqses_i) \text{ Si } r \leq 0,1 \\ \lambda(\text{sending}) = (out_2, reqses_i) \text{ Si } r \leq 0,2 \\ \lambda(\text{sending}) = (out_3, reqses_i) \text{ Si } r \leq 0,50 \\ \lambda(\text{sending}) = (out_4, reqses_i) \text{ Si } r \leq 0,725 \\ \lambda(\text{sending}) = (out_5, reqses_i) \text{ Si } r \leq 0,95 \\ \lambda(\text{sending}) = (out_6, reqses_i) \text{ Si } r \leq 1 \end{array} \right.$$

**Select Product:**

$$\text{Search} = \langle X_C, Y_C, S_C, P_C, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

*InputPorts* = {*in, fromCluster*} conjunto de puertos de entrada.

*OutputPorts* = {*out<sub>1</sub>, out<sub>2</sub>, out<sub>3</sub>, out<sub>4</sub>, toCluster*} conjunto de puertos de salida.

$$P_{SelectProduct} = \{(32,5\%, Browse, out_1), (32,5\%, Search, out_2), (30\%, Cart, out_3),$$

$$(5\%, End, out_4)\}$$

$$\lambda : \left\{ \begin{array}{l} r = \text{random U}[0,1] \\ \lambda(\text{sending}) = (out_1, reqses_i) \text{ Si } r \leq 0,325 \\ \lambda(\text{sending}) = (out_2, reqses_i) \text{ Si } r \leq 0,65 \\ \lambda(\text{sending}) = (out_3, reqses_i) \text{ Si } r \leq 0,95 \\ \lambda(\text{sending}) = (out_4, reqses_i) \text{ Si } r \leq 1 \end{array} \right.$$

**Modelo Atómico Transacción Pay**

En la Figura 4.7 se presenta el diagrama Statechart de la transacción Pay. Es similar a la del modelo atómico Transacciones con la incorporación del envío y recepción de

requerimientos al servicio Web.

En resumen su semántica consiste en (i) recibir los requerimientos por el puerto *in*, que pueden provenir del Experimental Framework como sesiones de clientes que se inician o de otra transacción, (ii) enviar los requerimientos recibidos por el puerto *in* a procesar a la plataforma tecnológica (*toCluster*); (iii) recibir requerimientos procesados por el puerto *fromCluster* y enviarlo al servicio Web por el puerto *toWS*, (iv) recibir requerimientos atendidos en el servicio Web, por el puerto *fromWS* y enviarlos a procesar a la plataforma informática y (v) decidir aleatoriamente el puerto de salida *out<sub>1</sub>*, *out<sub>2</sub>*, ..., *out<sub>n</sub>* a la próxima transacción.

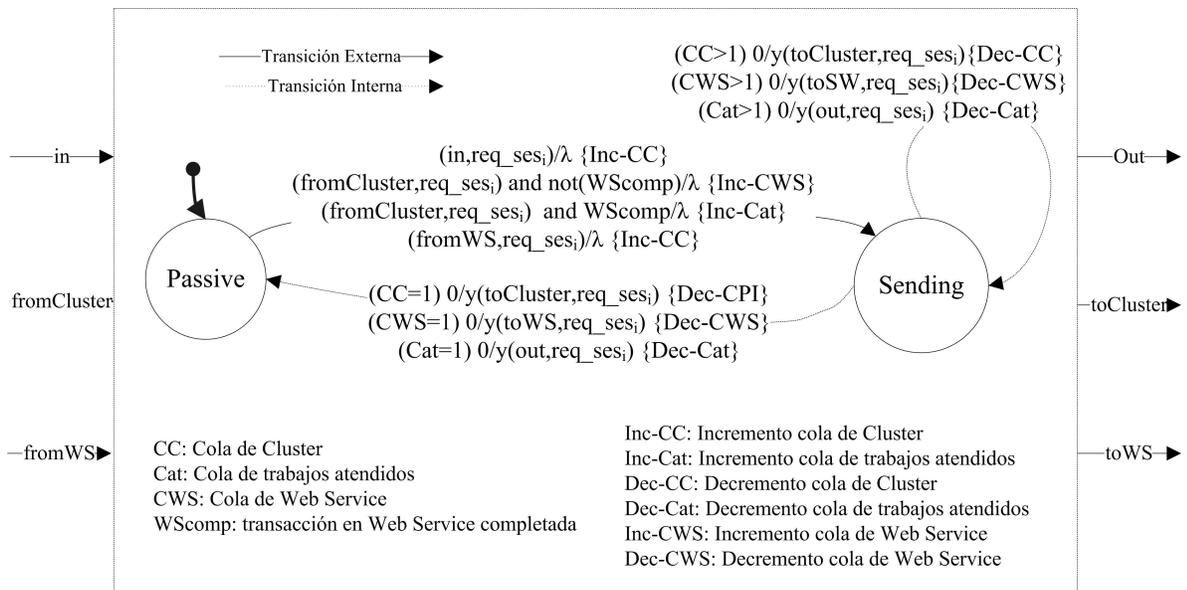


Figura 4.7: Diagrama Statechart del Modelo Atómico de Transacción Pay

El modelo atómico Pay se define como:

$$Pay = \langle X_{Pay}, Y_{Pay}, S_{Pay}, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

Conjuntos de puertos y valores permitidos

$$InputPorts = \{in, fromCluster, fromWS\} \text{ conjunto de puertos de entrada.}$$

$OutputPorts = \{out, toCluster, toWS\}$  conjunto de puertos de salida.

$X_{in} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de entrada  $in$  y representa los requerimientos a una transacción Pay.

$X_{fromCluster} = \{reqses_i\}$  es el conjunto de valores permitidos para el puerto de entrada  $fromCluster$  y representa los requerimientos procesados por el Cluster.

$X_{fromWS} = \{reqses_i\}$  es el conjunto de valores permitidos para el puerto de entrada  $fromWS$  y representa los requerimientos procesados por el servicio Web.

$Y_{out} = \{reqses_i\}$  conjunto de valores permitidos para el puerto de salida  $out$  y representa el envío de un requerimiento al Transducer.

$Y_{toCluster} = \{reqses_i\}$  conjunto de valores permitidos para el puerto de salida  $toCluster$  y representa el envío de un requerimiento para su proceso en el Cluster.

$Y_{toWS} = \{reqses_i\}$  conjunto de valores permitidos para el puerto de salida  $toWS$  y representa el envío de un requerimiento para su proceso en el servicio Web.

Conjuntos de eventos de entrada y salida.

$$X_{Pay} = \{(in, reqses_i), (fromCluster, reqses_i), (fromWS, reqses_i)\}$$

$$| in, fromCluster, fromWS \in InputPorts, reqses_i \in X_{in}, X_{fromCluster}, X_{fromWS}$$

$$Y_{Pay} = \{(out, reqses_i), (toCluster, reqses_i)\} | out, toCluster, toWS \in OutputPorts,$$

$$reqses_i \in Y_{out}, Y_{toCluster}, Y_{toWS}$$

Conjunto de estados.

$$S_{Pay} = \{passive, sending\}$$

Funciones:

$$\begin{aligned}
 \delta_{int} : & \begin{cases} \delta_{int}(sending) = sending \text{ Si } CC > 1 \\ \delta_{int}(sending) = passive \text{ Si } CC = 1 \\ \delta_{int}(sending) = sending \text{ Si } CWS > 1 \\ \delta_{int}(sending) = passive \text{ Si } CWS = 1 \\ \delta_{int}(sending) = sending \text{ Si } Cat > 1 \\ \delta_{int}(sending) = passive \text{ Si } Cat = 1 \end{cases} \\
 \delta_{ext} : & \begin{cases} \delta_{ext}(passive, e, (in, reqses_i)) = sending \\ \delta_{ext}(passive, e, ((fromCluster, reqses_i) \text{ and } not(WScomp))) = sending \\ \delta_{ext}(passive, e, ((fromCluster, reqses_i) \text{ and } WScomp)) = sending \\ \delta_{ext}(passive, e, (fromWS, reqses_i)) = sending \end{cases} \\
 \lambda(sending) &= (out, reqses_i) \\
 \lambda(sending) &= (toCluster, reqses_i) \\
 \lambda(sending) &= (toWS, reqses_i) \\
 \delta_{conf}(s, ta(s), x) &= \delta_{ext}(\delta_{int}, 0, x) \\
 ta : & \begin{cases} ta(passive) = \infty \\ ta(sending) = 0 \end{cases}
 \end{aligned}$$

## Modelo Atómico Web Service

En la Figura 4.8 se presenta el diagrama Statechart de la transacción Pay. Es similar a la del modelo atómico Transacciones con la incorporación del envío y recepción de requerimientos al servicio Web.

En resumen su semántica consiste en (i) recibir los requerimientos por el puerto *in* (ii) si está desocupado (*passive*) atenderlo, de lo contrario enviarlo a cola (*CaWS*) y (iii) una vez que el requerimiento es atendido se envía al Transducer por el puerto *out*.

Un modelo DEVS atómico de Web Service se representa por la estructura:

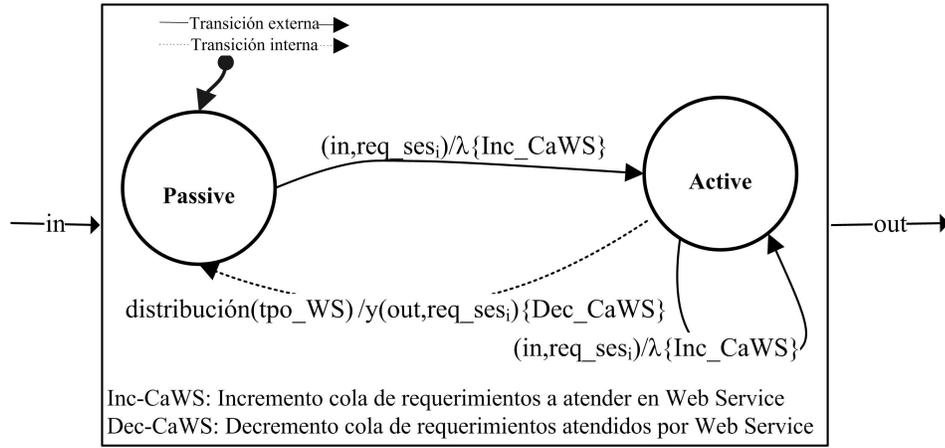


Figura 4.8: Diagrama Statechart del Modelo Atómico de Web Service

$$\text{Web Service} = \langle X_{WS}, Y_{WS}, S_{WS}, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

Conjuntos de puertos y valores permitidos.

$InputPorts = \{in\}$  conjunto de puertos de entrada.

$OutputPorts = \{out\}$  conjunto de puertos de salida.

$X_{in} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de entrada  $in$ .

$Y_{out} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de salida  $out$ .

Conjuntos de eventos de entrada y salida.

$X_{WS} = \{(in, reqses_i)\} \mid in \in InputPort \text{ y } reqses_i \in X_{in}$ .

$Y_{WS} = \{(out, reqses_i)\} \mid out \in OutputPort \text{ y } reqses_i \in Y_{out}$ .

Conjunto de estados.

$S = \{active, passive\}$  conjunto de estados.

Funciones

$$\delta_{int} : \begin{cases} \delta_{int}(active) = passive & \text{si CaWS} = 1 \\ \delta_{int}(active) = active & \text{si CaWS} > 1 \end{cases}$$

$$\delta_{ext} : \begin{cases} \delta_{ext}(passive, e, (in, reqses_i)) = active \\ \delta_{ext}(active, e, (in, reqses_i)) = active \end{cases}$$

$$\lambda(active) = Y(out, reqses_i)$$

$$\delta_{conf}(s, ta(s), x) = \delta_{ext}(\delta_{int}, 0, x)$$

$$ta : \begin{cases} ta(active) = \text{tiempo en servicio Web} \\ ta(passive) = \infty \end{cases}$$

### Modelo Acoplado Transaccional

Un modelo DEVS acoplado de transacciones queda definido como una estructura de elementos:

$$TransactionalModel = \langle X_{TM}, Y_{TM}, D, M_i, EIC, IC, EOC \rangle$$

$$X_{TM} = \{in, fromCluster\}$$

$$Y_{TM} = \{out, toCluster\}$$

$$D = \{VisitHomepage(VH), Browse(B), Search(S), SelectProduct(SP), Cart(C), \\ Pay(P), WebServicePay(WS)\}$$

$$M_i = \{M_{VH}, M_B, M_S, M_{SP}, M_C, M_P\}$$

$$EIC = \{((EF, out), (M_{VH}, in)), ((Cluster, out_1), (M_{VH}, fromCluster)), \\ ((Cluster, out_2), (M_B, fromCluster)), ((Cluster, out_3), (M_S, fromCluster)), \\ ((Cluster, out_4), (M_{SP}, fromCluster)), ((Cluster, out_5), (M_C, fromCluster)), \\ ((Cluster, out_6), (M_P, fromCluster))\}$$

$$\begin{aligned}
 IC &= \{((M_{VH}, out_1), (M_B, in)), ((M_{VH}, out_2), (M_S, in)), ((M_B, out_1), (M_S, in)), \\
 &\quad ((M_B, out_2), (M_B, in)), ((M_B, out_3), (M_{SP}, in)), ((M_S, out_1), (M_B, in)), \\
 &\quad ((M_S, out_2), (M_S, in)), ((M_S, out_3), (M_{SP}, in)), ((M_C, out_1), (M_{SP}, in)), \\
 &\quad ((M_C, out_2), (M_C, in)), ((M_C, out_3), (M_P, in)), ((M_C, out_4), (M_B, in)), \\
 &\quad ((M_C, out_5), (M_S, in)), ((M_P, out_1), (M_{WS}, in)), ((M_{WS}, out), \\
 &\quad (M_P, fromWS))\} \\
 EOC &= \{((M_{VH}, toCluster), (Cluster, in)), ((M_B, toCluster), (Cluster, in)), \\
 &\quad ((M_S, toCluster), (Cluster, in)), ((M_{SP}, toCluster), (Cluster, in)), \\
 &\quad ((M_C, toCluster), (Cluster, in)), ((M_P, toCluster), (Cluster, in)), \\
 &\quad ((M_B, out_4), (EF, toTransducer)), ((M_S, out_4), (EF, toTransducer)), \\
 &\quad ((M_{SP}, out_4), (EF, toTransducer)), ((M_C, out_6), (EF, toTransducer)), \\
 &\quad ((M_P, out_1), (EF, toTransducer))\}
 \end{aligned}$$

### 4.7.3. Modelo de Plataforma Tecnológica

Figura 4.9 se muestra el diagrama Statechart de la plataforma tecnológica con la consideración de fallas en los servidores.

El modelo Plataforma Tecnológica consta de un puerto de entrada *in* que recibe los requerimientos del modelo transaccional y fallas, mientras que el puerto *out* retorna los requerimientos atendidos. Está compuesto por los modelos atómicos Coordinator que cumple la función de un balanceador de cargas y Server. Coordinator posee los puertos de entrada *in* y *fromServer*.

Como puertos de salida se consideran  $n$  puertos *toServer<sub>j</sub>*, uno por cada objeto servidor, así como un puerto *out*. El modelo atómico Servidor puede instanciarse en  $n$  objetos de la clase y consta de los puertos *in* y *out*. El puerto *in* recibe requerimientos o fallas y retorna por el puerto *out* los requerimientos atendidos y el fin de una falla

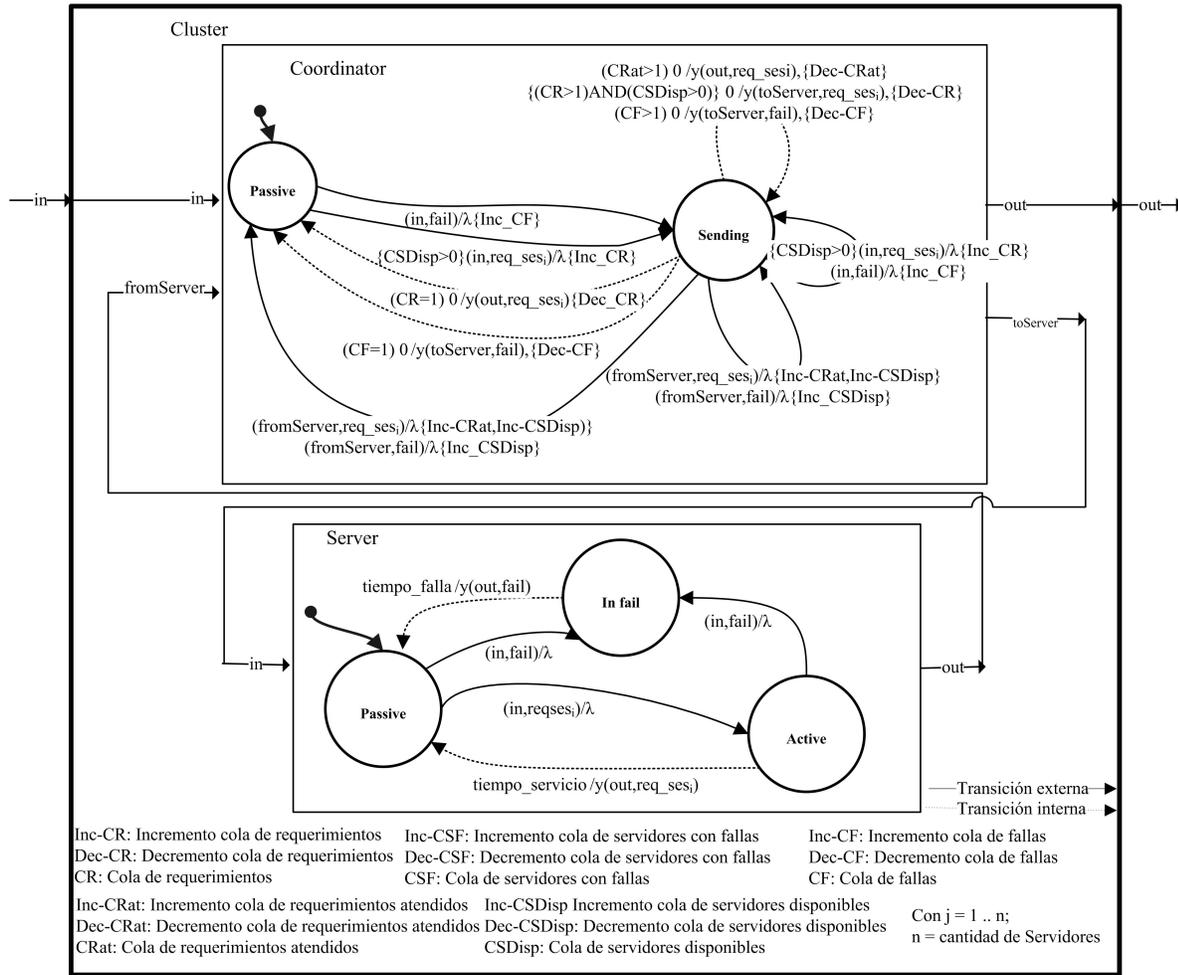


Figura 4.9: Diagrama Statechart del Modelo de Plataforma Tecnológica con Fallas

(servidor recuperado).

Para el caso del balanceador de carga se especifica el tratamiento de las colas. Las mismas son de tipo FIFO y contienen requerimientos arribados, requerimientos atendidos, servidores disponibles, fallas y servidores con fallas.

De acuerdo a la transición ejecutada se incrementa o decrementa la cola correspondiente. Para comprender la interpretación de una transición se explican dos casos del modelo atómico servidor:

- **Transición interna  $tiempoServicio/(out, reqSes_i)$ :** cumplido un *tiempo de servicio*

se produce la salida del requerimiento atendido por el puerto *out*.

- **Transición externa**  $(in, reqSes_i)/\lambda$ : se produce una entrada de un requerimiento por el puerto *in* y se deja expresado con  $\lambda$  la imposibilidad de producir salidas.

### Modelo Atómico Coordinator

Un modelo DEVS atómico Coordinator con cola FIFO se representa por la estructura:

$$\text{Coordinator} = \langle X_C, Y_C, S, \delta_{int}, \delta_{ext}, \delta_{conf}, \lambda, ta \rangle$$

Conjuntos de puertos y valores permitidos.

$InputPorts = \{in\}$  conjunto de puertos de entrada.

$OutputPorts = \{out\}$  conjunto de puertos de salida.

$X_{in} = Req \cup Fails$  es el conjunto de valores permitidos para el puerto de entrada *in*, que se forma de la unión de los conjuntos de requerimientos arribados con el de fallas.

$Y_{out} = Req \cup Fails$  es el conjunto de valores permitidos para el puerto de salida *out*, que se forma de la unión de los conjuntos de requerimientos arribados con el de fallas.

Conjuntos de eventos de entrada y salida.

$X_C = \{(in, reqses_i), (in, fail_i)\} \mid in \in OutputPorts, reqses_i \in Req \text{ y } fail_i \in Fails.$

$Y_C = \{(out, reqses_i), (out, fail_i)\} \mid out \in OutputPorts \text{ y } reqses_i \in Req \text{ y } fail_i \in Fails.$

Conjunto de estados.

$S = \{passive, sending\}$  conjunto de estados.

Funciones

$$\delta_{int} : \begin{cases} \delta_{int}(sending) = passive & \text{si CR} = 1 \\ \delta_{int}(sending) = sending & \text{si CR} > 1 \end{cases}$$

$$\begin{aligned}
\delta_{ext} : & \begin{cases} \delta_{ext}(passive, e, (in, reqses_i)) = sending \\ \delta_{ext}(sending, e, (in, reqses_i)) = sending \\ \delta_{ext}(passive, e, (in, fail_i)) = sending \\ \delta_{ext}(sending, e, (in, fail_i)) = sending \end{cases} \\
\lambda : & \begin{cases} \lambda(sending) = Y(out, fail_i) \text{ si } \exists fail \\ \lambda(sending) = Y(out, reqses_i) \text{ en otro caso} \end{cases} \\
\delta_{conf}(s, ta(s), x) &= \delta_{ext}(\delta_{int}, 0, x) \\
ta : & \begin{cases} ta(sending) = 0 \\ ta(passive) = \infty \end{cases}
\end{aligned}$$

### Modelo Atómico Server

Un modelo DEVS atómico Server sin cola se representa por la estructura:

$$\text{Server} = \langle X_s, Y_s, S, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

Conjuntos de puertos y valores permitidos.

$InputPorts = \{in\}$  conjunto de puertos de entrada.

$OutputPorts = \{out\}$  conjunto de puertos de salida.

$X_{in} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de entrada  $in$ .

$Y_{out} = \{reqses_i\}$  con  $i = 1..n$  es el conjunto de valores permitidos para el puerto de salida  $out$ .

Conjuntos de eventos de entrada y salida.

$X_s = \{(in, reqses_i)\} \mid in \in InputPorts \text{ y } reqses_i \in X_{in}$ .

$Y_s = \{(out, reqses_i)\} \mid out \in OutputPorts \text{ y } reqses_i \in Y_{out}$ .

Conjunto de estados.

$S = \{passive, active\}$

Funciones

$$\begin{aligned} \delta_{int} &: \left\{ \begin{array}{l} \delta_{int}(active) = passive \\ \delta_{ext}(passive, e, (in, reqses_i)) = active \end{array} \right. \\ \lambda(active) &= Y(out, reqses_i) \\ ta: &\left\{ \begin{array}{l} ta(active) = \text{tiempo de servicio} \\ ta(passive) = \infty \end{array} \right. \end{aligned}$$

**Modelo Acoplado del Cluster**

$$\text{Cluster} = \langle X_c, Y_c, S, D, M_i, EIC, IC, EOC \rangle$$

$$X_c = \{in\}$$

$$Y_c = \{out\}$$

$$D = \{Coordinator, Server\}$$

$$M_i = \{M_{Coordinator}, M_{Server}\}$$

$$EIC = \{(Coordinator, in), (Cluster, in)\}$$

$$IC = \{(((Coordinator, out), (Cluster, in)), ((Cluster, out), (Coordinator, in)))\}$$

$$EOC = \{((Coordinator, out), (Cluster, out))\}$$

## 4.8. Obtención de los Parámetros del Modelo

La adquisición de los parámetros se realiza a través de la medición de tiempos de servicios de cada transacción mediante una herramienta de monitorización.

SoapUI (<http://www.soapui.org>) es un monitor por software de libre licencia y código abierto con capacidades de testeo de aplicaciones. Cuenta con una interface gráfica de fácil utilización para evaluación de cargas y simulación (mock services).

Para obtener los parámetros de los recursos informáticos se proponen tres escenarios, el primero con una configuración de baja capacidad, el segundo con una de media capacidad y en el tercero una de alta capacidad.

En la Tabla 4.2 se presenta la configuración de los recursos informáticos, organizado en tres escenarios de acuerdo a sus características.

Descripción	Escenario 1	Escenario 2	Escenario 3
Procesador	Intel Pentium 1,33 GHz.	Intel Pentium 4,3 GHz.	AMD Phenom II
Cantidad de Núcleos	1	2	4
Memoria RAM	DDR 1, 1 GB.	DDR 1, 2 GB	DDR 3, 4 GB.
Disco Rígido	SATA 120 GB.	SATA 320 GB	2 de 320 GB RAID 1
Costo por Unidad	U\$D 300	U\$D 400	U\$D 720

Tabla 4.2: Configuración de los recursos

Sobre la base de un proyecto con el WSDL de las operaciones de servicios Web para cada una de las transacciones electrónicas, se simula la ejecución en un servidor en cada configuración informática de los escenarios, se hacen requerimientos y se miden los tiempos de servicio demandados.

En la Figura 4.10 se muestra la ventana del software SoapUI para la medición de las transacciones del sitio. En la misma se observa un proyecto ejemplo en el cual está en ejecución un servidor Web con las transacciones buy, login, logout y search.

El cobro electrónico se implementa a través de una asociación a un servicio contratado a Paypal, cuya operación consiste en enviar los datos del comprador en un mensaje SOAP para que se ejecute el cobro. En la Figura 4.11 se muestra la ventana principal con el proyecto Paypal abierto.

De las operaciones del proyecto Paypal se selecciona DoDirectPayment para su evaluación (Figura 4.12).

Para la ejecución de la medición se crea un requerimiento (Request) a ser ejecutado (Figura 4.13). Por cada requerimiento se obtiene el tiempo de servicio de la operación.

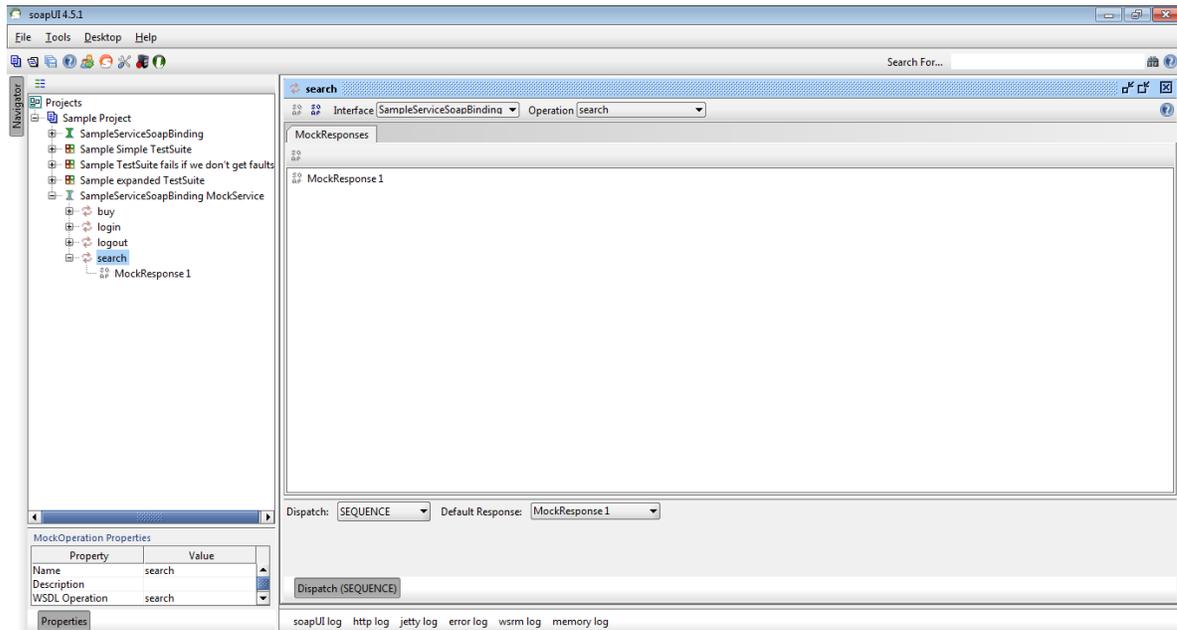


Figura 4.10: Ventana principal soapUI

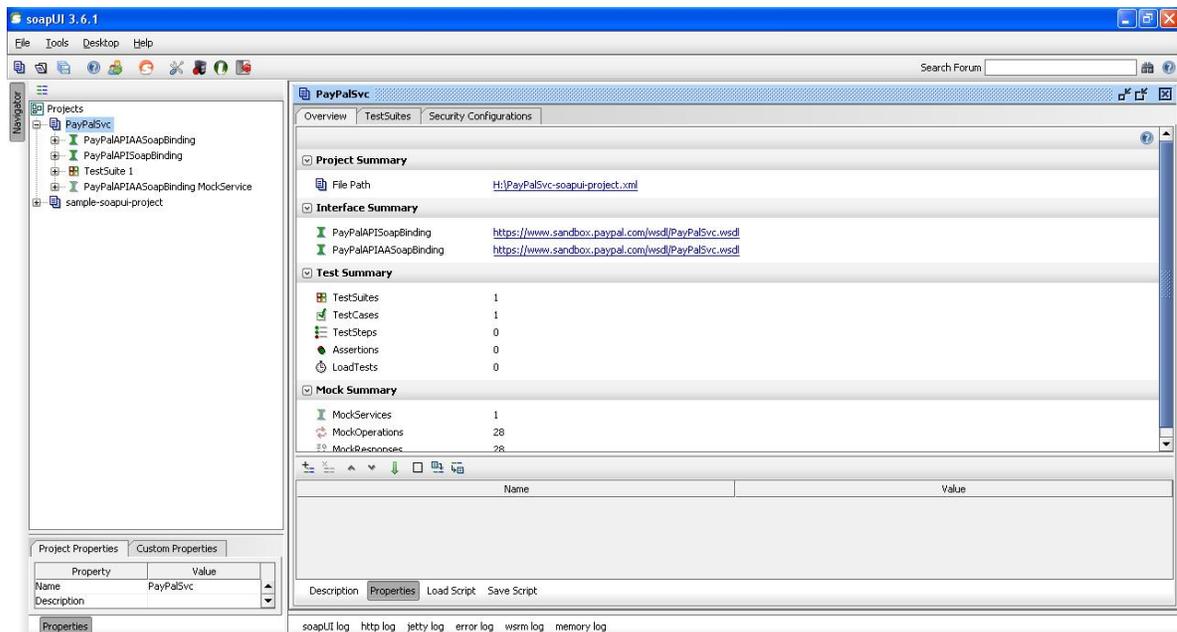


Figura 4.11: Ventana principal soapUI para el proyecto Paypal

Con su uso se mide el tiempo de ejecución de un pago directo y de sus experimentaciones se calcula el tiempo promedio de servicio en la transacción de pago. Este tiempo

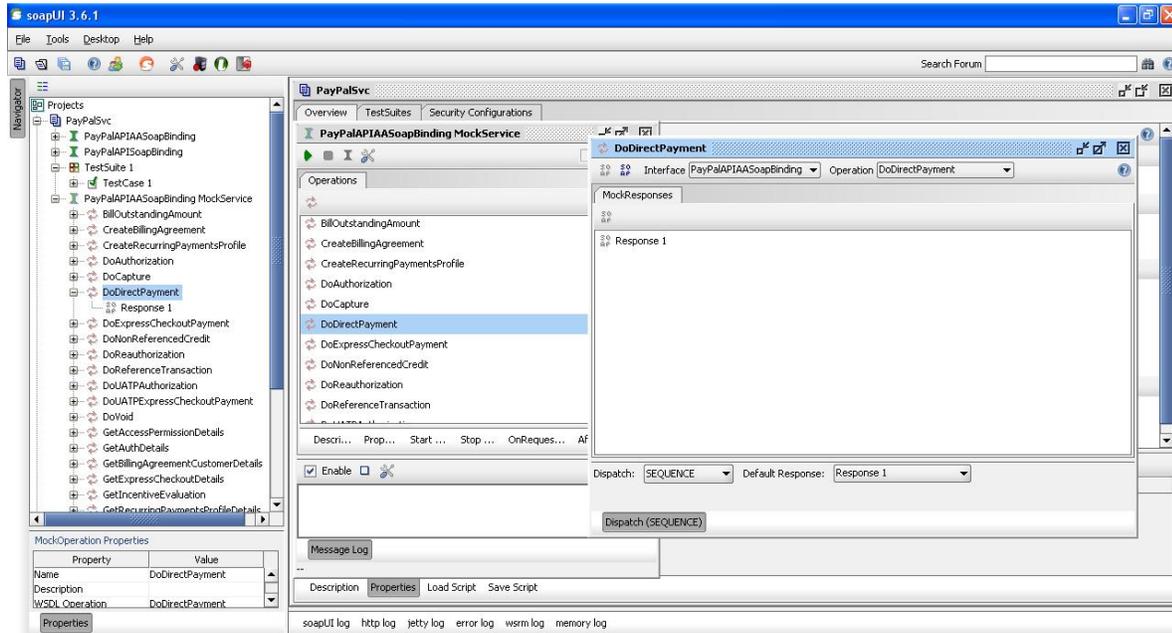


Figura 4.12: Ventana de la operación Direct Payment

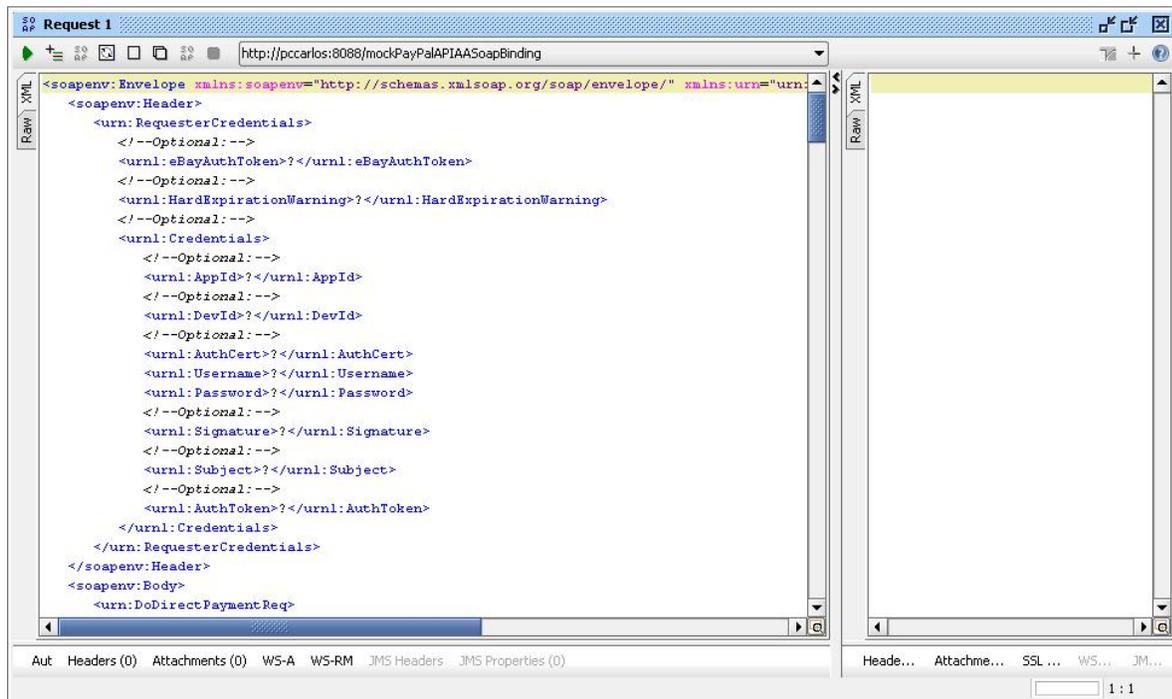


Figura 4.13: Ventana Request de la operación Direct Payment

es de 5.43 segundos.

Se ejecutan las experimentaciones y de su monitorización se obtienen los tiempos de servicio. En la Tabla 4.3 se presentan los tiempos de servicios promedio para el Escenario 1, en la Tabla 4.4 los tiempos de servicios promedio para el Escenario 2 y en la Tabla 4.5 los correspondientes al Escenario 3.

Transacción	Tiempo de Servicio
Visit Home Page	1.951
Search	0.027
Browse	0.013
Select	0.021
Cart	0.025
Pay	0.022

Tabla 4.3: Tiempo de Servicio por Transacción para Escenario 1

Transacción	Tiempo de Servicio
Visit Home Page	0,0218
Serach	0,0078
Browse	0,0125
Select	0,0062
Cart	0,0315
Pay	0,0185

Tabla 4.4: Tiempo de Servicio por Transacción para Escenario 2

Transacción	Tiempo de Servicio
Visit Home Page	0.0978
Serach	0.0035
Browse	0.0086
Select	0,0054
Cart	0.0025
Pay	0.0074

Tabla 4.5: Tiempo de Servicio por Transacción para Escenario 3

Siguiendo el trabajo de referencia de Menascé y otros (2000) se consideran las siguientes cantidades de clientes arribados por segundo: 20, 15, 10, 5 y 1, con una distribución exponencial del tiempo entre arribos. En la Tabla 4.6 se detallan los tiempos entre arribos para cada tasa.

Tasa de arribos (clientes/segundos)	Tiempo entre arribos (segundos)
20	0,05
15	0,067
10	0.1
5	0.2
1	1

Tabla 4.6: Cargas de clientes al Sitio

## 4.9. Implementación en DEVSJAVA

Para comprender los fundamentos de implementación se representa el modelo de comercio electrónico en un diagrama UML en la Figura 4.14.

Cada modelo atómico es parte de la clase `ViewableAtomic` y los acoplamientos se describen en la clase `ViewableDigraph`.

`Experimental Framework` es un modelo acoplado formado por los modelos atómicos `Generator` y `Transducer`, `Transactional Model` es un modelo acoplado de  $n$  transacciones atómicas y `Cluster Model` es el modelo acoplado correspondiente a la capa informática formado por un modelo atómico coordinador (`Coordinator`) y  $n$  modelos atómicos de servidores (`Server`).

En la Figura 4.15 se profundiza el detalle del diagrama de clases del modelo DEVS de comercio electrónico. En el mismo se observa una clase correspondiente al modelo acoplado de comercio electrónico `CMElectronicCommerce`, de la cual son parte los modelos acoplados `CMExperimentalFramework`, `CMTransactions` y `CMTechnologicalPlatform`.

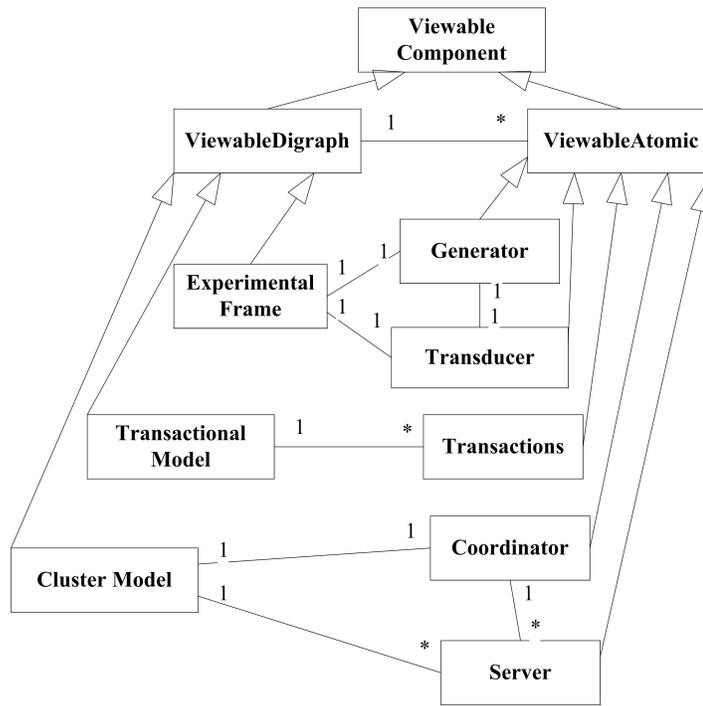


Figura 4.14: Diagrama de Clases del Modelo de Comercio Electrónico

CMExperimentalFramework se forma por las clases Generator y Transducer. De CMTransactions son parte tres tipos de transacciones: las que son parte del proceso interno de negocios y que se comportan como centros de cola (ProcQ); las que además del proceso interno invocan a un servicio Web (PrecInvWS) y las que proveen el servicio Web (ProcWS).

Por último, el CMTechnologicalPlatform compuesta por las clases Coordinator y Server. La plataforma tecnológica puede sufrir fallas lo que se indica con la clase atómica FailGenerator.

El modelo DEVSJAVA se muestra en la Figura 4.16 y consiste en: (i) modelo acoplado Experimental Framework que genera los arribos al sitio, monitoriza el comportamiento y computa las métricas de salida, (ii) Modelo Transaccional que consiste en una red acoplada de transacciones de tipo plana con probabilidades de transiciones entre ellas, (iii) Cluster es un modelo acoplado jerárquico de plataforma informática de

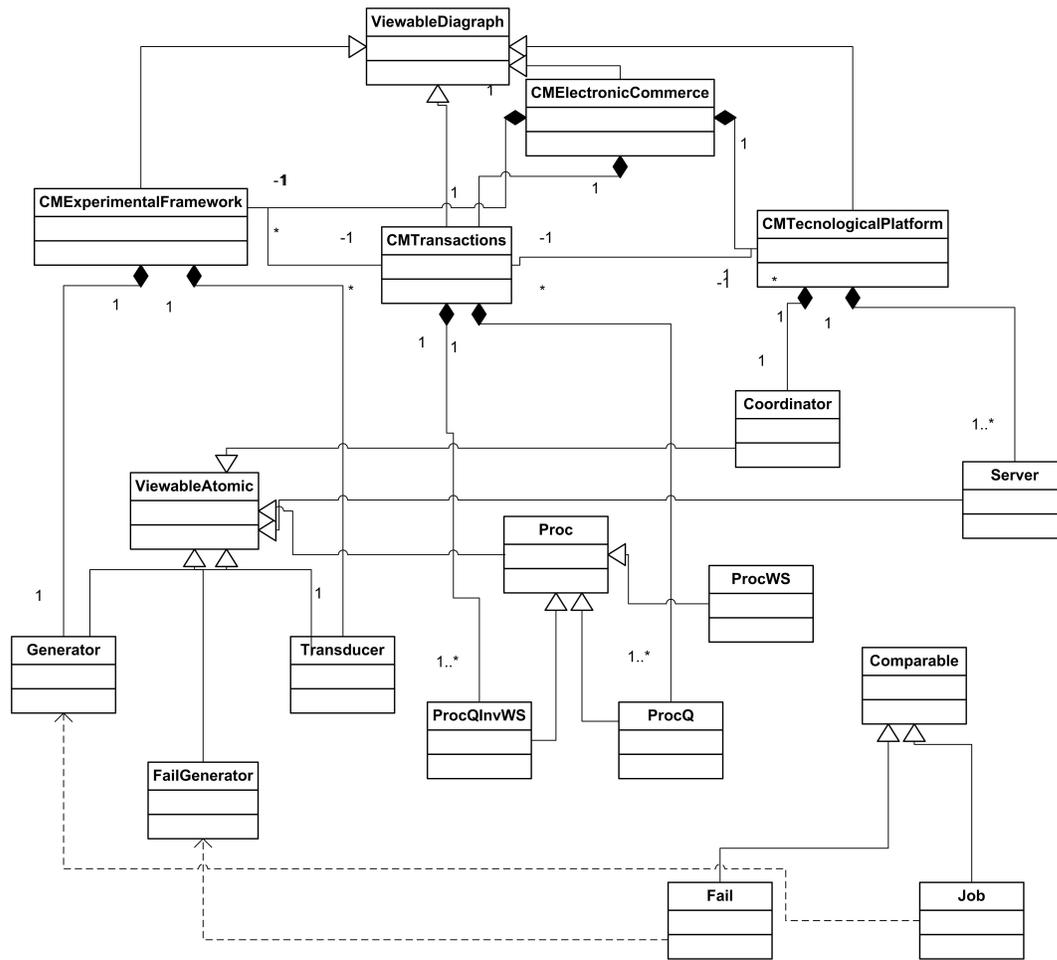


Figura 4.15: Diagrama de Clases del Modelo DEVS

un balanceador de carga y los servidores Web, (iv) un modelo atómico asociado a la transacción Pagar que se corresponde con el servicio de ejecución del pago electrónico.

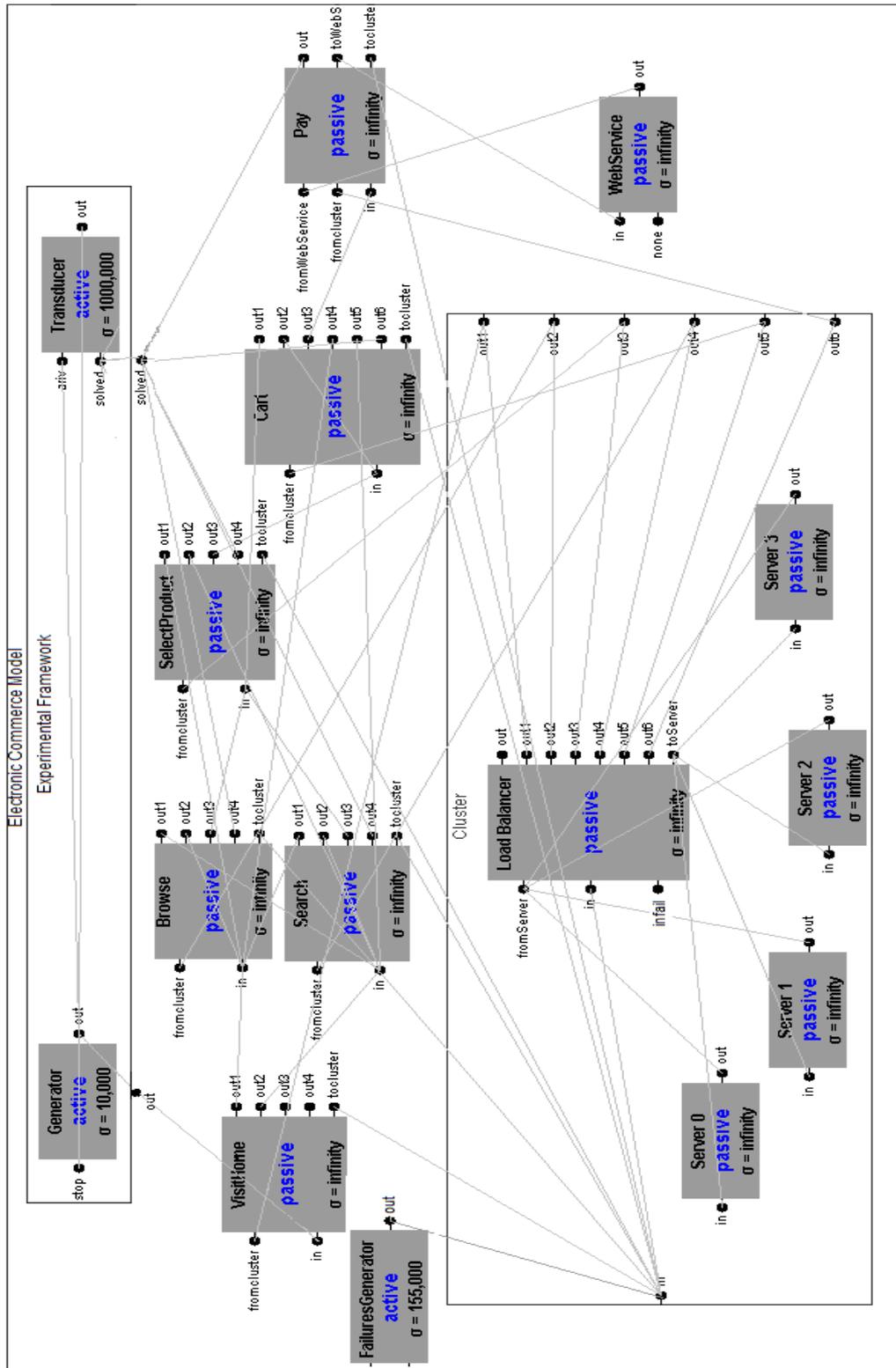


Figura 4.16: Ventana Simulador DEVJAVA Modelo de Comercio Electrónico.

## 4.10. Verificación del Modelo de Simulación

La verificación del modelo se instrumenta en base a dos técnicas:

1. **Rastreo de a pasos de mensajes:** utilización de la opción *Step* del Simview de DEVJAVA para la ejecución paso a paso de la simulación y seguimiento de mensajes.
2. **Uso de archivos de Log:** generación de archivos de Log y cálculo de métricas para comparación con las resultantes de simulación.

Rastreando los mensajes entre componentes del modelo se verifica desde su construcción. En forma incremental, se implementa el modelo con acoplamientos simples, verificando la correcta ejecución del programa desde un análisis lógico.

Para el proceso de verificación del modelo propuesto se plantea un tiempo entre arribos constante de 10 segundos y un cluster con diez servidores.

Las simulaciones se ejecutan en una computadora personal con procesador Pentium Dual-Core de 2,7 Ghz, 2 Gb. de memoria RAM y disco rígido de 459 Gb.

Para este caso se establece un tiempo de corrida de veinte mil segundos lo cual demanda veinte minutos en ejecución.

De la corrida de simulación se obtienen los siguientes archivos de Log, a partir de los cuales se calculan las métricas:

1. **Generador:** nombre sesión, tiempo de inicio de sesión, tiempo entre arribos.
2. **Recorrido:** nombre sesión, tiempo de finalización de sesión, secuencia de transacciones de cada sesión.

### 4.10.1. Cálculo de Tiempo de Respuesta del Sistema

Por cada sesión se calcula el *tiempo de finalización de sesión* (tomado del archivo Recorrido) menos el *tiempo inicio de sesión* (tomado del archivo Generador).

Una vez realizados los cálculos para cada sesión se obtiene el promedio de los mismos, el cual se lo compara con el tiempo de respuesta obtenido en la simulación.

En la Tabla 4.7 se presentan los resultados del tiempo de respuesta y se concluye que verifican la correcta obtención de la métrica.

Promedio obtenido por:	Tiempo de Respuesta (segundos)
Simulación	6,9147
Datos Log	6,9953

Tabla 4.7: Tiempos de Respuesta

### 4.10.2. Cálculo de la Velocidad de Procesamiento del Sistema

A partir del archivo de log Recorrido se obtiene la cantidad de sesiones completadas. Se divide el número de sesiones finalizadas sobre el tiempo de simulación. Este resultado obtenido se lo compara con la velocidad de procesamiento de la simulación.

En la Tabla 4.8 se presentan los resultados de la velocidad de procesamiento y se concluye que verifican la correcta obtención de la métrica.

Promedio obtenido por:	Velocidad de Procesamiento (req./seg.)
Simulación	0,1
Datos Log	0,0999

Tabla 4.8: Velocidad de Procesamiento

## 4.11. Aplicación del Algoritmo de Configuraciones Eficientes

### 4.11.1. Realizar una Planificación de Capacidad.

Para planificar la capacidad de los recursos se necesita simular con los parámetros planteados hasta encontrar una plataforma informática que muestre un balance ente desempeño tecnológico y costos. Como medida de desempeño tecnológico se toma el Tiempo de Respuesta.

La plataforma tecnológica a simular varía en el número de servidores Web del cluster, el cual comienza en uno y se incrementa hasta llegar a 7 servidores.

Se simula el proceso de comercio electrónico para los tres escenarios de acuerdo a las cargas de trabajos con una distribución exponencial de los tiempo entre arribos: 0.05, 0.067, 0.1, 0.2 y 1 segundo (Tabla 4.6).

Para encontrar un costo de simulación razonable se ejecutan pruebas y se analiza el tiempo para el cual el sistema produce valores estables de las métricas.

En este caso se observa que para 11000 segundos el comportamiento del sistema es estable, lo cual demanda 15 minutos en promedio de tiempo de simulación.

Por cada experimentación se realizan dos repeticiones y por tanto para cada escenario se ejecutan setenta corridas con un tiempo estimado de 17.5 horas de simulación con una demora 52.5 horas de simulación para los tres escenarios.

### 4.11.2. Identificar Recursos Críticos

El análisis se inicia simulando el modelo para el Escenario 1. En la Figura 4.17 se grafican las diferentes cantidades de requerimientos en cola de servidores.

Se observa una saturación del sistema para tiempos entre arribos de 0.05 seg., 0.067 seg., 0.1 seg. y 0.2 seg. Para el caso de 1 seg. de tiempo entre arribos el sistema está saturado para uno y dos servidores, con una disminución a veinte requerimientos en cola para tres servidores, diez para cuatro servidores y nueve para cinco, seis y siete servidores.

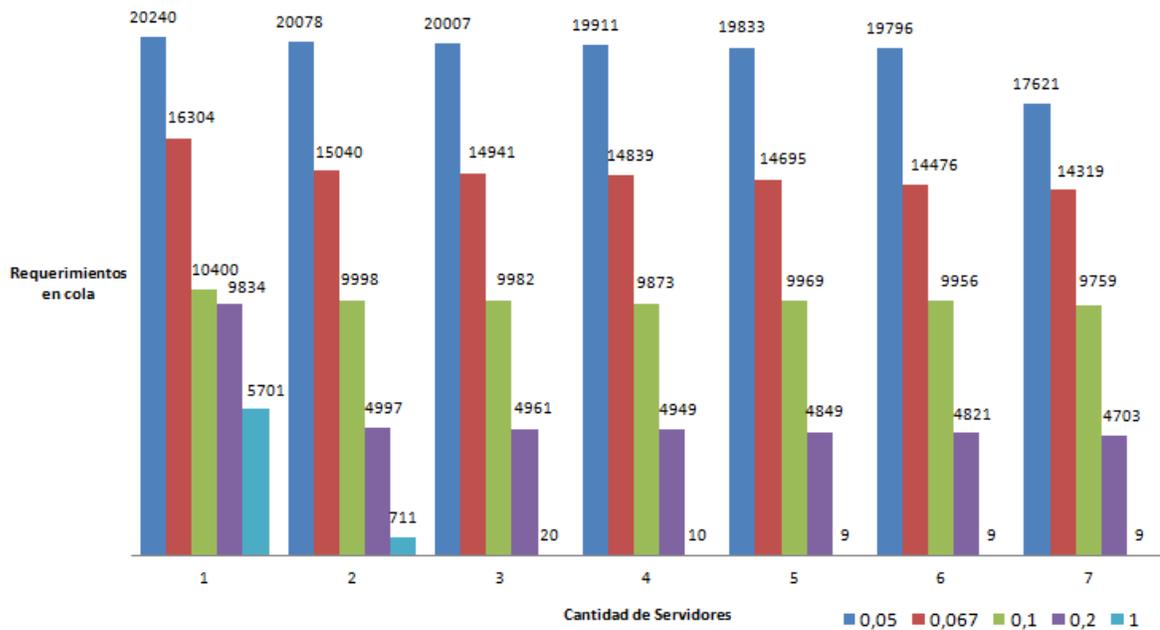


Figura 4.17: Cantidad de requerimientos en cola para diferentes cantidades de servidores en Escenario 1.

Para completar el análisis del escenario 1, se estudian los tiempos de respuesta del sistema, graficados en la Figura 4.18. Se observa que no se alcanzan valores adecuados del indicador del tiempo de respuesta.

Por tanto el escenario 1 no es una plataforma tecnológica eficiente ya que no alcanza tiempos de respuesta deseados.

En la Figura 4.19 se muestran las colas para el escenario 2, se observa que se logra disminuir el número de requerimientos en cola para un servidor con tiempo entre arribos de un 1 seg. y para todos los tiempos entre arribos simulados, con siete servidores.

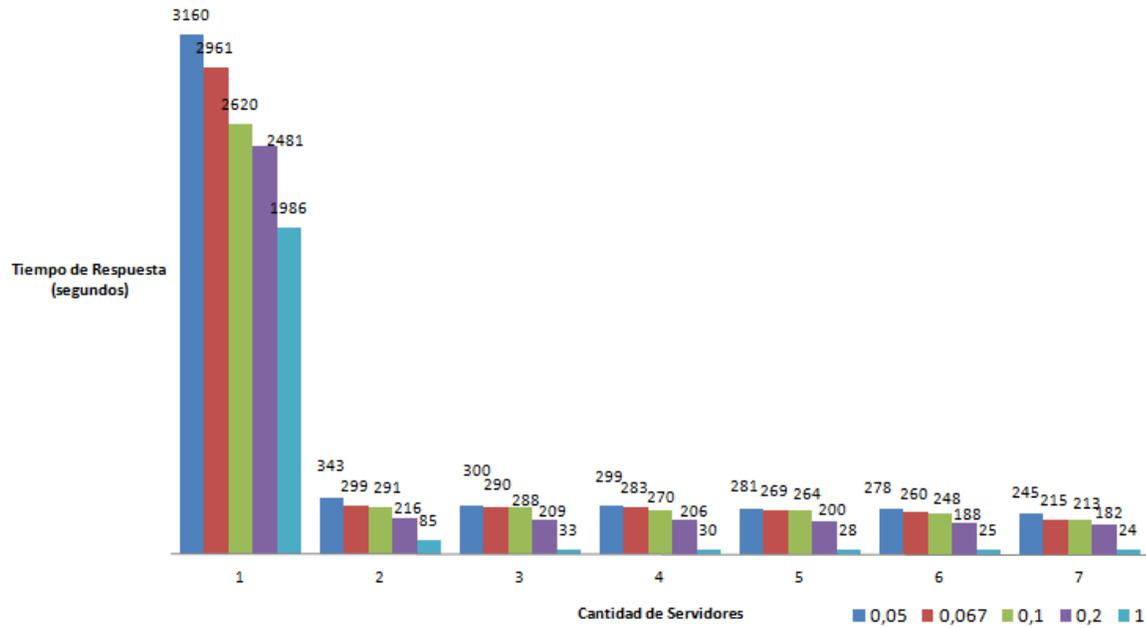


Figura 4.18: Tiempo de Respuesta para diferentes cantidades de servidores en Escenario 1.

Estando el resto de los casos en saturación.

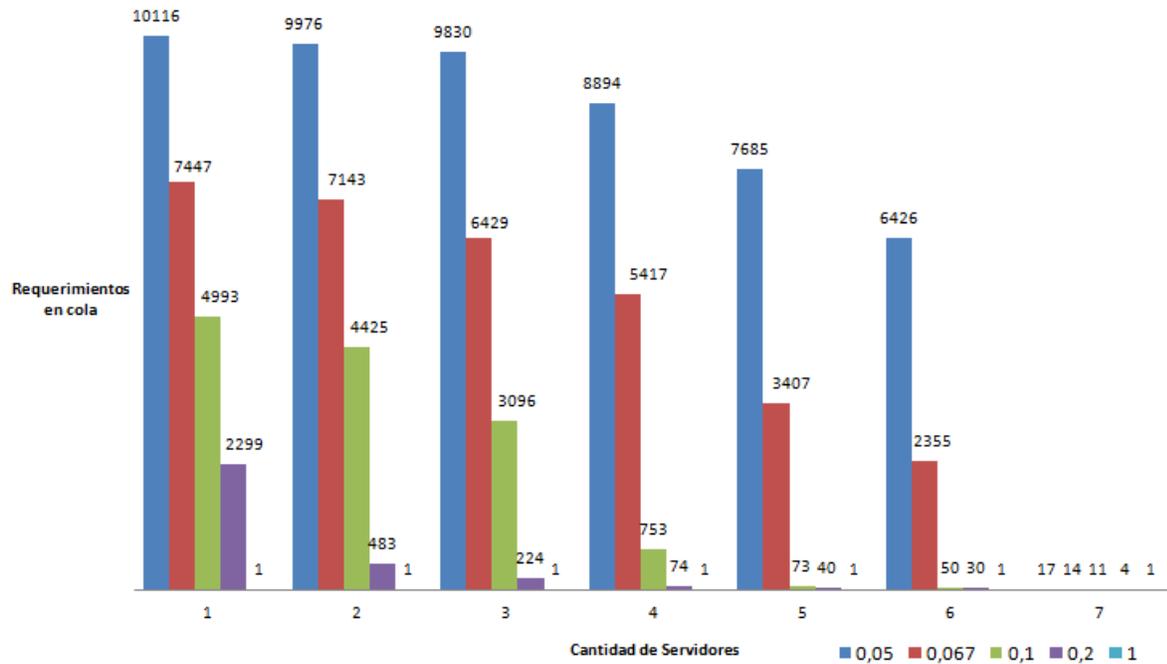


Figura 4.19: Cantidad de requerimientos en cola para diferentes cantidades de servidores en Escenario 2.

Los tiempos de respuesta para el escenario 2 se presentan en la Figura 4.20. Se observa que para siete servidores se logran tiempos de respuesta *Sobresalientes* para 0.1, 0.2 y 1 seg. de tiempo entre arribos y *Satisfactorio* para 0.05 y 0.067 seg. de tiempo entre arribos.

Para seis servidores se obtiene tiempo de respuesta *Sobresaliente* para 1 seg. de tiempo entre arribos y *Satisfactorio* para 0.1 y 0.2 seg. de tiempo entre arribos.

Por último para cinco servidores se alcanzan tiempos de respuestas *Satisfactorios* para 0.2 y 1 seg. de tiempo entre arribos.

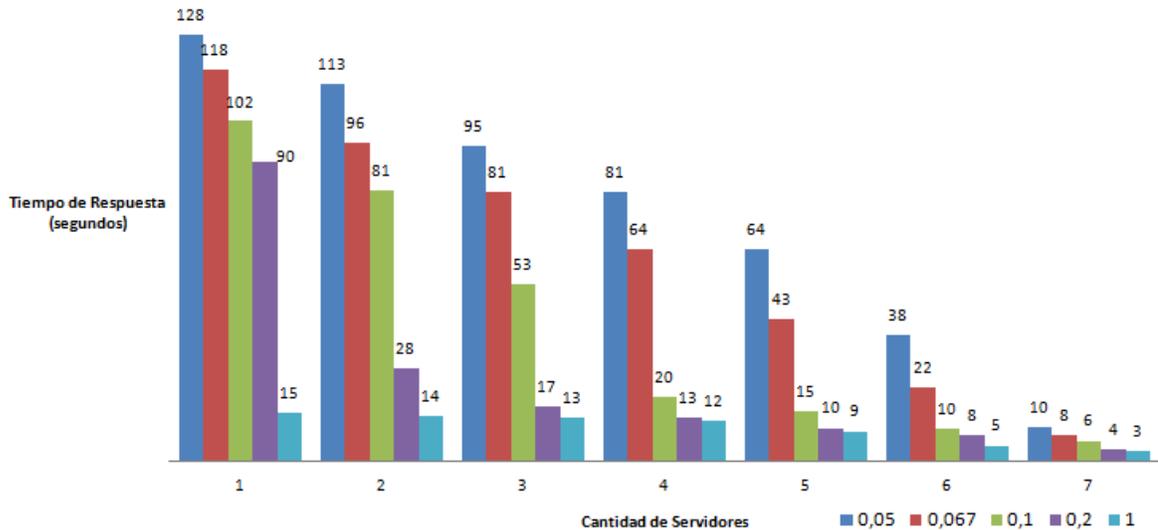


Figura 4.20: Tiempo de Respuesta para diferentes cantidades de servidores en Escenario 2.

Si bien en el escenario 2 se obtienen casos con tiempos de respuesta adecuados al indicador, se pueden analizar nuevas configuraciones que con la disminución de servidores e incremento de su capacidad se mejoren los resultados.

En la Figura 4.21 se muestra las colas de requerimientos en el escenario 3, la cual muestra una reducción de requerimientos en cola, manteniendo la saturación para los casos de tres servidores con tiempos entre arribos de 0.05 y 0.067 seg., dos servidores con tiempos entre arribos de 0.05, 0.067 y 0.1 seg. y un servidor con tiempos entre

arribos de 0.05, 0.067, 0.1 y 0.2 seg.

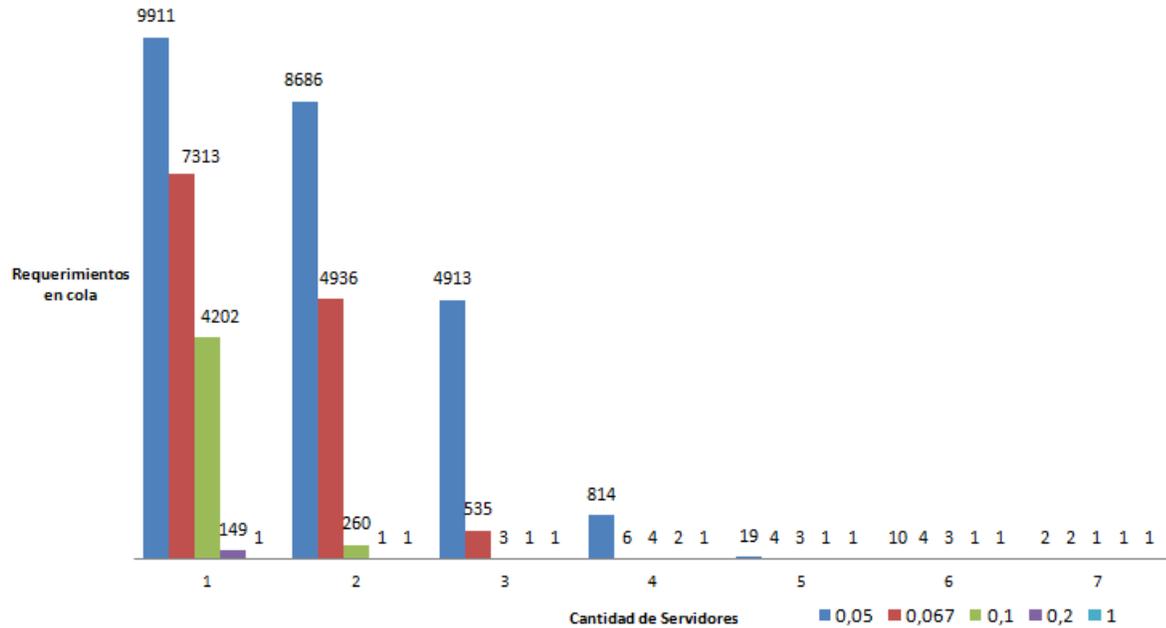


Figura 4.21: Cantidad de requerimientos en cola para diferentes cantidades de servidores en Escenario 3.

En la Figura 4.22 se grafican los tiempos de respuesta para el escenario 3, en la cual se observa que con cinco, seis y siete servidores se logran valores *Sobresalientes* del tiempo de respuesta, con cuatro servidores se alcanza valores *Sobresalientes* para 0.1, 0.2 y 1 seg. del tiempo entre arribos y *Suficientes* para 0.05 y 0.067 seg. del tiempo entre arribos.

Con tres servidores no se llega a obtener valores adecuados del indicador para 0.05 y 0.067 seg. del tiempo entre arribos.

Desde la perspectiva tecnológica se concluye que con cuatro servidores se obtienen indicadores del tiempo de respuesta a niveles *Sobresalientes* y *Suficientes*, siendo para cinco servidores todos *Sobresalientes*.

Para tomar la decisión se requiere del análisis económico, a partir del cual se estudia la posibilidad de incrementar el número de servidores para disponer de mejor tecnología.

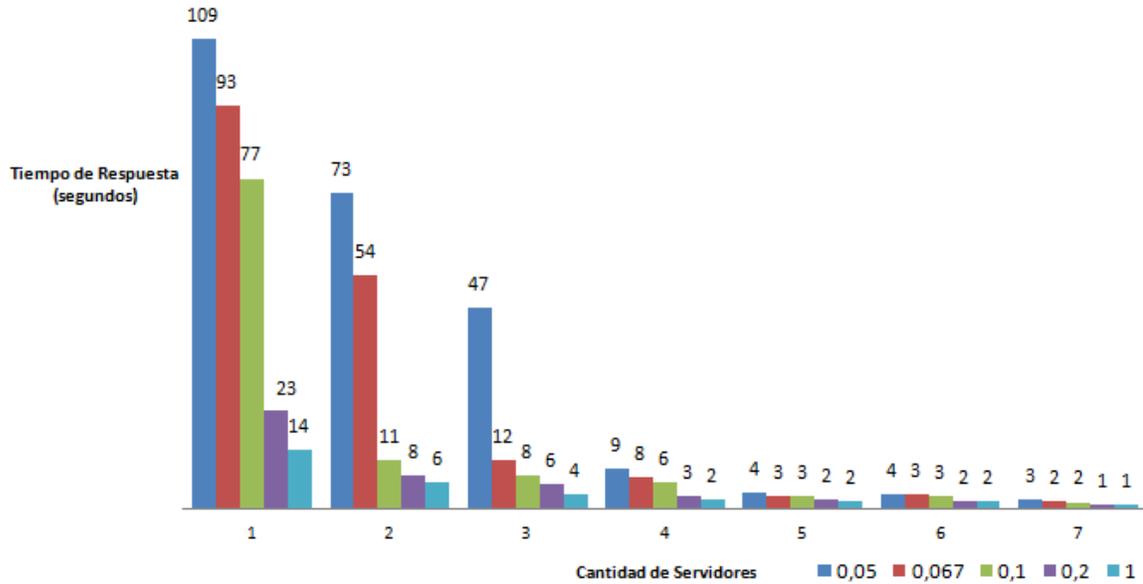


Figura 4.22: Tiempo de Respuesta para diferentes cantidades de servidores para Escenario 3.

### 4.11.3. Análisis del Indicador Económico

La métrica de negocios consiste en obtener un punto de equilibrio entre el costo de la inversión y los beneficios netos. Para ello se calcula el costo de inversión para cinco servidores del escenario 3, como la suma de los costos de la plataforma informática y del desarrollo del sistema de comercio electrónico.

En la Tabla 4.9 se muestra que el costo total de inversión para el escenario 3, con 5 servidores es de U\$D 23.600.

Concepto	Monto (U\$D)
5 servidores	3600
Costos del desarrollo del sitio de negocios electrónicos	20.000
<b>Costo Total de Inversión</b>	<b>23.600</b>

Tabla 4.9: Costos de Inversión Escenario 3 con cinco servidores

Para un tiempo entre arribos de 0.05 seg. se obtienen 54 clientes en promedio por

día. Considerando una venta promedio de 1620 productos al mes, con un precio de venta de U\$D 30 por libro se obtiene un ingreso por ventas de U\$D 48.600. Se necesita saber cuál es el ganancia neta obtenida. De acuerdo a la Ecuación 2.10:

$$\text{Ganancia Neta} = \text{U\$D } 48,600 * \text{Porcentaje de Beneficio}$$

De la aplicación de la Ecuación 2.11 con datos del Estado de Resultados de la empresa para diciembre de 2012, se obtiene un *Procentaje de Beneficio* de 5.86 %.

Por tanto la Ganancia Neta es:

$$\text{Ganancia Neta} = \text{U\$D } 48,600 * 5,86 \% = \text{U\$D } 2,847,96$$

Por tanto se necesitan 8 meses de retorno de la inversión para alcanzar el punto de equilibrio, 9,5 para lograr una utilidad adecuada y 10 para buena utilidad.

Los meses a partir de los cuales obtienen beneficios muestran que es factible invertir en la plataforma informática con cinco servidores. Además si se considera el costo de inversión para cuatro servidores su disminución respecto a la alternativa de 5 servidores es de U\$D 720, lo cual no incide significativamente en el tiempo de retorno de la inversión.

## 4.12. Diseño Experimental Estadístico

Se propuso la evaluación del rendimiento de la plataforma tecnológica en relación con la productividad de negocios del proceso de comercio electrónico.

De la aplicación del algoritmo de configuraciones eficientes se concluye que desde la perspectiva integrada de negocios y tecnología es conveniente experimentar sobre las siguientes variables de diseño tecnológico:

- **Cantidad de servidores del cluster:** cluster compuesto por cuatro servidores para alcanzar los indicadores y cluster con cinco servidores para lograr los indicadores y contar con capacidad como proyección de crecimiento.
- **Servicio Web de Pago:** la mejora en los tiempos de servicio por parte de la empresa responsable del cobro electrónico.
- **Fallas:** posibilidad de incorporar la incidencia de fallas en el sistema.

Para estas alternativas se calcula la cantidad de clientes que compran como medida de negocios.

Por tanto el objetivo del experimento es analizar el efecto de la cantidad de servidores, la mejora en los tiempos de servicio Web de pago y las fallas en las variables de productividad tecnológica y de negocios.

Como se necesita obtener la cantidad de productos vendidos por día se toma como tamaño de muestra por experimentación una cantidad de tiempo que posibilite dicho cálculo.

Respecto al tiempo de simulación se analizan resultados de monitorización de sitios de comercio electrónico y se observa que el mayor número de clientes se encuentra entre las 8 h. y las 20 h. De modo que se simula por un tiempo de 12 horas.

En la Tabla 4.10 se presentan los factores y sus niveles, los cuales consisten en la configuración de la plataforma informática con la variante de cuatro o cinco servidores, el nivel disponibilidad con la incorporación de fallas y diferentes tiempos del servicio Web de pago.

Las variables de respuesta se proponen en la Tabla 4.11 y son el tiempo de respuesta y la velocidad de procesamiento desde la perspectiva tecnológica y la cantidad de clientes que compran desde la de negocios.

Factores	Niveles
Configuración tecnológica	Escenario 3 con 4 servidores
	Escenario 3 con 5 servidores
Disponibilidad	Con fallas
	Sin fallas
Tiempo Servicio Web	5.43
	4.43
	3.43

Tabla 4.10: Factores y niveles del experimento

Grupo	Variable de Respuesta
Tecnológicas	Tiempo de Respuesta
	Velocidad de Procesamiento
Negocios	Cantidad de productos vendidos

Tabla 4.11: Variables de Respuesta

### 4.12.1. Diseño Multifactorial Completo

Se considera un diseño multifactorial completo, para un modelo de efectos fijos, con el fin de estudiar el efecto de los factores y sus interacciones en las variables de respuesta. Se efectúan cuatro repeticiones por nivel y se simula un tiempo de ocho horas por experimentación.

En la herramienta de software Minitab, licencia 4829180127571000656, se implementa el diseño experimental estadístico para la obtención del análisis de varianza y de los gráficos de salidas.

Con los resultados de simulación se debe comprobar la idoneidad del modelo mostrando el cumplimiento de los supuestos sobre los residuos (Sección 2.22).

### 4.12.2. Comprobación de la Idoneidad del Modelo para el Tiempo de Respuesta

Del gráfico 4.23 se concluye:

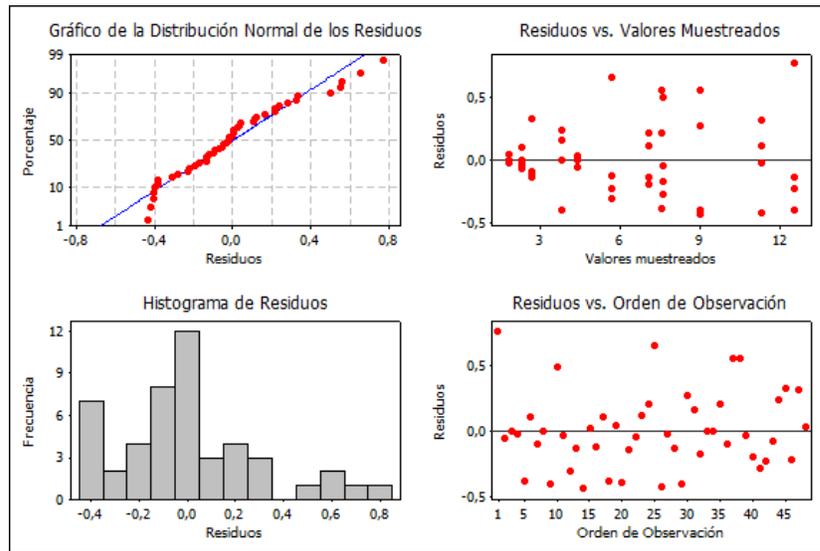


Figura 4.23: Residuos para el Tiempo de Respuesta en el Diseño Factorial Completo.

- **Suposición de Normalidad:** de la figura se puede observar que el gráfico de probabilidad normal no muestra residuos inusitados y a su vez el 100% de los residuos estandarizados se encuentran entre  $\pm 0,5$  y los mismos están alineados aproximadamente a la recta de 45 grados. Además el histograma de residuos muestra una tendencia a la distribución Normal.
- **Gráfico de residuos versus el orden de los datos observados en el tiempo:** no se observa correlación entre los residuos, ya que una tendencia a rachas de residuos positivos y negativos indica una correlación positiva y esto implica que se viola la suposición de independencia de los residuos. Por ello se muestra un procedimiento apropiado de aleatorización en la experimentación.
- **Gráfico de residuos versus el valor ajustado (valor promedio del nivel):** como no se observa un patrón se asegura la varianza constante.

### 4.12.3. Comprobación de la Idoneidad del Modelo para la Velocidad de Procesamiento

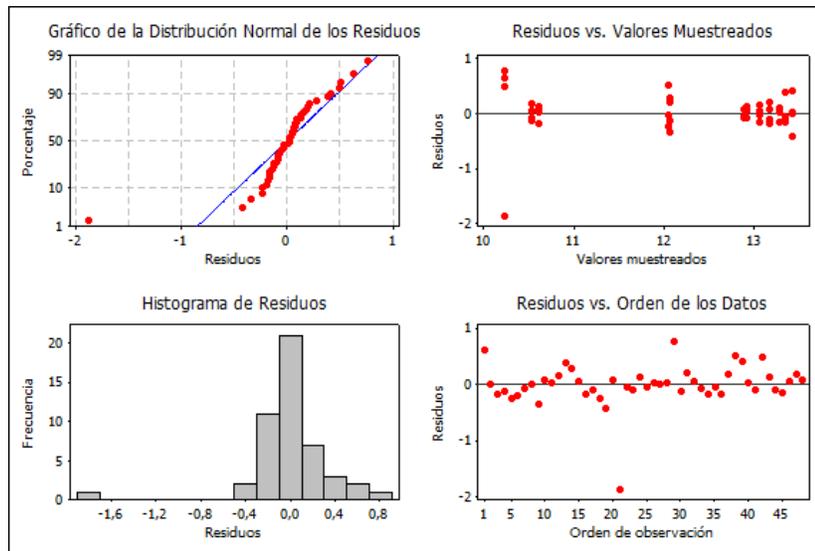


Figura 4.24: Residuos para la Velocidad de Procesamiento en el Diseño Factorial Completo.

Del gráfico 4.24 se concluye:

- **Suposición de Normalidad:** de la figura se puede observar que el gráfico de probabilidad normal no muestra residuos inusitados y a su vez el 100 % de los residuos estandarizados se encuentran entre  $\pm 0,5$  y los mismos están alineados aproximadamente a la recta de 45 grados. Además el histograma de residuos muestra una tendencia a la distribución Normal.
- **Gráfico de residuos versus el orden de los datos observados en el tiempo:** no se observa correlación entre los residuos, por ello se observa un procedimiento apropiado de aleatorización en la experimentación.
- **Gráfico de residuos versus el valor ajustado (valor promedio del nivel):** como no se observa un patrón se asegura la varianza constante.

#### 4.12.4. Análisis de Resultados de Simulación

Se realiza un Diseño Factorial Completo con un modelo de efectos fijos cuyos datos se detallan en la Tabla 4.12.

Se simula por un tiempo de veinte mil segundos en un equipo con procesador Pentium Dual-Core de 2,7 Ghz, 2 Gb. de memoria RAM y disco rígido de 459 Gb., con una demora de diez minutos para la ejecución.

Factor	Tipo	Niveles	Valores
Tecnología de Servidores	fijo	2	Esc3Serv4. Esc3Serv5
Disponibilidad	fijo	2	SinFallas. ConFallas
Tiempo Servicio Web	fijo	3	3,43. 4,43. 5,43

Tabla 4.12: Datos para el experimento

- **Número de Experimentos** = Niveles Tecnología Servidores \* Niveles Disponibilidad \* Niveles Servicio Web \* Número de Repeticiones.
  - **Número de Experimentos** =  $2 * 2 * 3 * 4 = 48$
- **Tiempo de Experimentación** = Número de Experimentos \* Tiempo Promedio de Simulación por Experimento.
  - **Tiempo de Experimentación** =  $48 * 2 \text{ [h]} = 96 \text{ [h]}$

Por tanto si se dedican cinco horas diarias se estiman veinte días de trabajo de simulación.

##### 4.12.4.1. Análisis de Varianza para la Variable de Respuesta Tiempo de Respuesta

En la Tabla 4.13 se muestra el Análisis de Varianza para la variable de respuesta tiempo de respuesta y se determina la significación de cada factor e interacción.

Factores/Interacciones	GL	SC	MC	$F$	$P$	$F_{Tabla}$	Significación
Tecnología Servidores (TServ)	1	188,03	188,030	1706,25	0	4,113	Significativo
Disponibilidad (D)	1	135,23	135,23	1227,15	0	4,113	Significativo
Tiempo Servicio Web (TSW)	2	176,81	88,40	802,23	0	3,259	Significativo
TServ * D	1	4,69	4,69	42,61	0	4,113	Significativo
TServ * TSW	2	26,57	13,28	120,58	0	3,259	Significativo
D * TSW	2	0,001	0,001	0	0,995	3,259	No significativo
TServ * D * TSW	2	3,50	1,75	15,88	0	4,113	Significativo
Error	36	3,96	3,96	0,110			
Total	47	538,81					

Tabla 4.13: ANOVA de los factores e interacciones para la variable de respuesta Tiempo de Respuesta. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados.

Del análisis de varianza se concluye que no es significativa la interacción de la Disponibilidad y el Servicio Web ya que su valor  $p$  es mayor que  $\alpha$  y el  $F$  calculado es menor que el  $F$  de la tabla. Siendo significativos los factores Tecnología de Servidores, Disponibilidad y Tiempo de Servicio Web, así como las interacciones Tecnología de Servidores y Disponibilidad, Tecnología de Servidores y Tiempo de Servicio Web y la interacción de los tres factores. Por tanto la selección de los factores es relevante ya que la incidencia en la variable de respuesta es significativa.

En el gráfico 4.25 se presentan los tiempos de respuesta promedio para los factores en estudio.

Se alcanzan valores *Sobresalientes* del indicador para la configuración con cinco servidores, con y sin la existencia de fallas y para los tres tiempos de servicio Web.

Para el caso de cuatro servidores se obtiene un valor *Sobresaliente* para el menor tiempo de servicio Web y sin fallas, *Satisfactorio* para un tiempo de servicio Web de 4.43 y 5.43 seg. sin fallas y 4.43 seg. con fallas. El resto de los casos es *Insatisfactorio*.

Se puede afirmar entonces que la disminución del tiempo de servicio demandado por el servicio Web de Paypal es conveniente para la mejora en el tiempo de respuesta, así como contar con cinco servidores en el cluster.

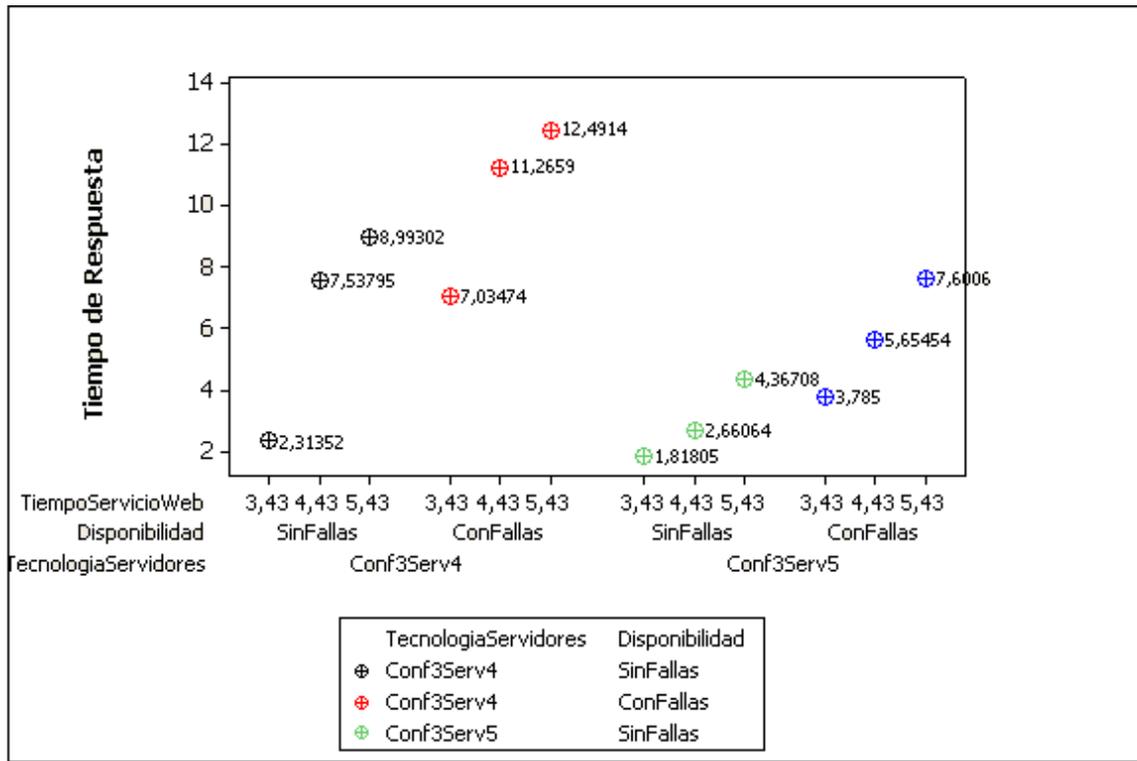


Figura 4.25: Tiempo de Respuesta versus factores en el Diseño Factorial Completo.

En el gráfico 4.26 se representan los efectos de los factores frente a un cambio de nivel. Se observa que el factor cantidad de servidores es sensible a los cambios de niveles, de modo que al incrementar el número de servidores, se decrementa el tiempo de respuesta. La Disponibilidad muestra sensibilidad y por ello al incorporar fallas se produce un incremento del tiempo de respuesta.

Para el caso del tiempo de Servicio Web también se muestra sensibilidad a los cambios de nivel, de modo que al incrementar el mismo se decrementa el tiempo de respuesta.

En el gráfico 4.27 se presentan las interacciones entre los factores. De la interpretación de tecnología de servidores se observa un incremento en el tiempo de respuesta producido por la existencia de fallas y la disminución de cinco a cuatro servidores. Respecto a la disponibilidad se demuestra su efecto negativo por la existencia de fallas y

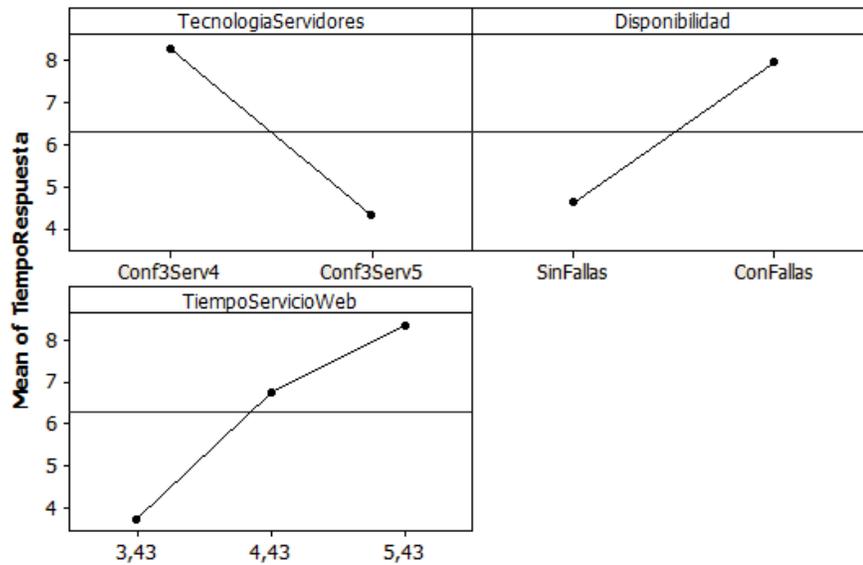


Figura 4.26: Efectos de los factores para el Tiempo de Respuesta en el Diseño Factorial Completo.

menor cantidad de servidores.

Por último para los diferentes tiempos de servicio Web se analiza un decremento en el tiempo de respuesta a medida que se disminuye el tiempo de servicio Web y se evitan fallas.

Por tanto se concluye que para mejorar el tiempo de respuesta es conveniente incorporar cinco servidores al cluster y lograr un tiempo en servicio Web de Paypal de 3.43 seg. Si se logra esta plataforma se consiguen tiempos de respuesta *Sobresalientes* a pesar de la posibilidad de fallas.

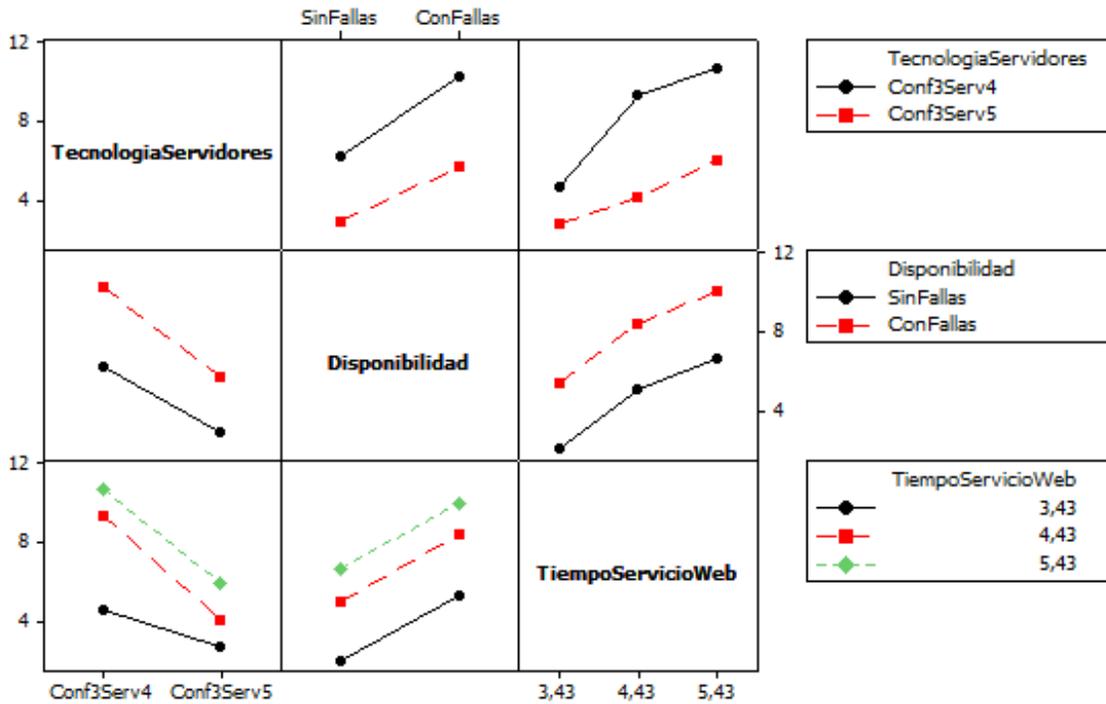


Figura 4.27: Interacciones de los factores para el Tiempo de Respuesta en el Diseño Factorial Completo.

#### 4.12.4.2. Análisis de Varianza para la Variable de Respuesta Velocidad de Procesamiento

En la Tabla 4.14 se muestra el análisis de varianza para la variable de respuesta velocidad de procesamiento y se determina la significación de cada factor e interacción.

Son significativos los factores Tecnología de Servidores, Disponibilidad, Tiempo de Servicio Web y la interacción Tiempo de Servicio Web \* Disponibilidad en la variable de respuesta Velocidad de Procesamiento. Caso contrario para las interacciones: Tecnología de servidores \* Tiempo de Servicio Web, Disponibilidad \* Tiempo de Servicio Web y Tecnología de servidores \* Tiempo de Servicio Web \* Disponibilidad.

Por tanto se demuestra que los factores propuestos en el diseño experimental y la

Factores/Interacciones	GL	SC	MC	$F$	$P$	$F_{Tabla}$	Significación
Tecnología Servidores (TServ)	1	38,24	38,24	218,53	0	4,113	Significativo
Disponibilidad (D)	1	14,29	14,29	81,67	0	4,113	Significativo
Tiempo Servicio Web (TSW)	2	1,41	0,7071	4,04	0,026	3,259	Significativo
TServ * D	1	7,29	7,29	41,66	0	4,113	Significativo
TServ * TSW	2	0,42	0,21	1,21	0,311	3,259	No Significativo
D * TSW	2	0,28	0,14	0,82	0,446	3,259	No significativo
TServ * D * TSW	2	0,35	0,1778	1,02	0,372	3,259	No Significativo
Error	36	6,30	6,30	0,175			
Total	47	68,61					

Tabla 4.14: ANOVA de los factores e interacciones para la variable de respuesta Velocidad de Procesamiento. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados.

interacción Tiempo de Servicio Web y Disponibilidad son significativos en la velocidad de procesamiento, mientras que el resto de las interacciones no afectan a la mencionada métrica.

En el gráfico 4.28 se presentan las velocidades de procesamiento para los diversos factores del diseño experimental.

Se alcanzan las máximas velocidades de procesamiento para cinco servidores, sin fallas y los diferentes tiempos de servicios Web. En segundo lugar, cinco servidores con fallas, tercero cuatro servidores sin fallas y con los menores velocidades de procesamientos cuatro servidores con fallas.

En el gráfico 4.29 se observa mayor sensibilidad de la velocidad de procesamiento frente a los cambios de tecnología y disponibilidad. Siendo mínima para el caso de cambio en el tiempo de servicio Web.

En el gráfico 4.30 se muestran las interacciones de los factores, del cual se observa que contar con cinco servidores, minimizando las fallas logra las mayores cantidades de requerimientos por segundo, no afectando el tiempo de servicio Web a la métrica en estudio.

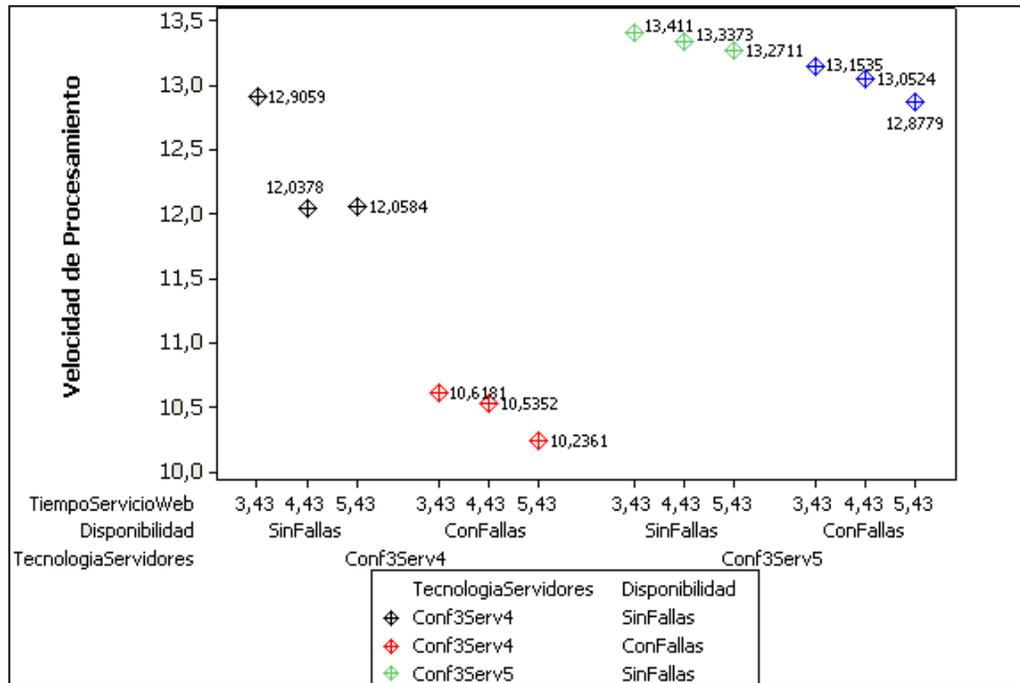


Figura 4.28: Velocidad de Procesamiento versus factores en el Diseño Factorial Completo.

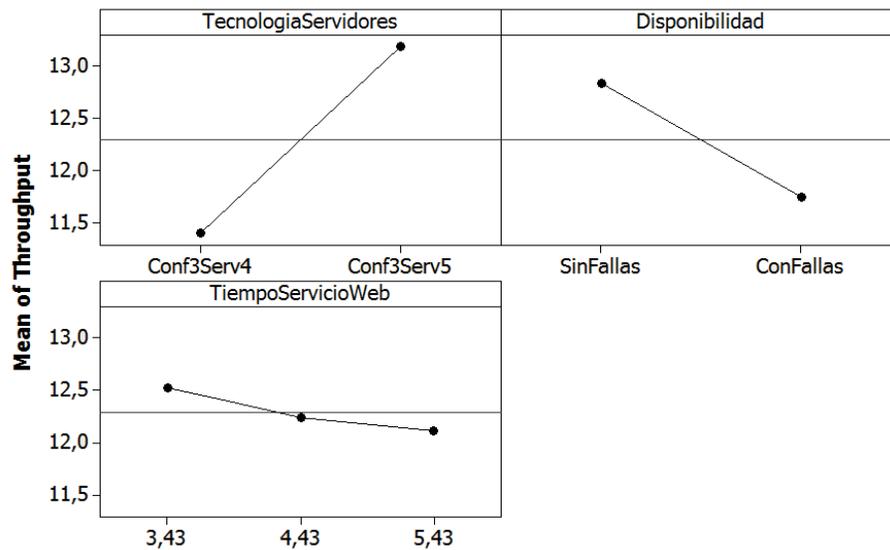


Figura 4.29: Efectos de los factores para la Velocidad de Procesamiento en el Diseño Factorial Completo.

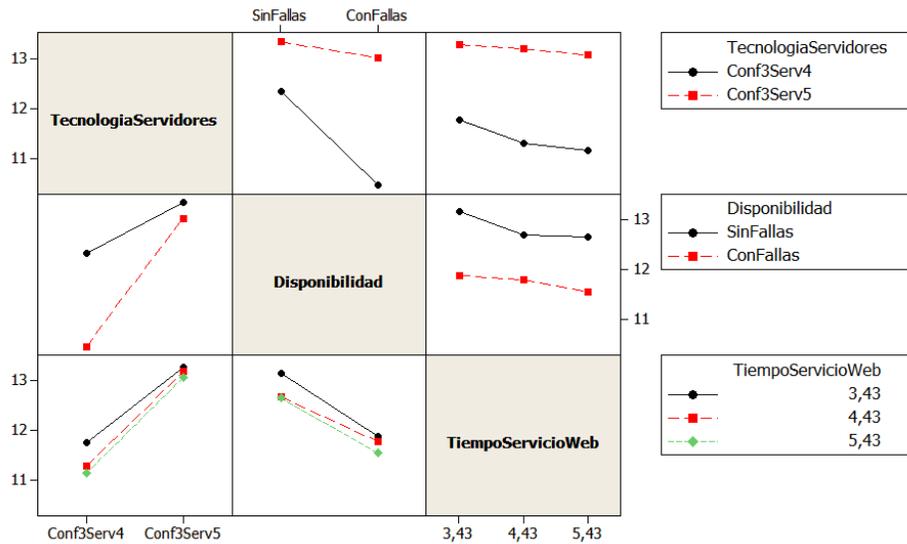


Figura 4.30: Interacciones de los factores para la Velocidad de Procesamiento en el Diseño Factorial Completo.

En la Tabla 4.15 se presentan las desviaciones estándares de la velocidad de procesamiento y tiempo de respuesta de los datos del diseño experimental estadístico.

TecnologiaServidores	Disponibilidad	Tiempo Serv. Web	Veloc. Proc.	T. Resp.
Conf. 3, 4 servidores	con fallas	3.43	1,14	0,19
Conf. 3, 4 servidores	con fallas	4.43	2,09	0,28
Conf. 3, 4 servidores	con fallas	5.43	2,55	0,72
Conf. 3, 4 servidores	sin fallas	3.43	1,01	0,08
Conf. 3, 4 servidores	sin fallas	4.43	1,29	0,50
Conf. 3, 4 servidores	sin fallas	5.43	1,14	0,44
Conf. 3, 5 servidores	con fallas	3.43	1,18	0,54
Conf. 3, 5 servidores	con fallas	4.43	1,34	0,49
Conf. 3, 5 servidores	con fallas	5.43	1,58	0,53
Conf. 3, 5 servidores	sin fallas	3.43	1,35	0,03
Conf. 3, 5 servidores	sin fallas	4.43	1,26	0,32
Conf. 3, 5 servidores	sin fallas	5.43	1,13	0,04

Tabla 4.15: Desviación Estándar del Diseño Experimental Estadístico

### 4.12.5. Métricas de Negocios

En el gráfico 5.21 se presentan las cantidades de productos vendidos por día para los factores del diseño experimental, de la cual se observa que los máximos resultados se obtienen para el menor tiempo de servicio Web de 3.43 seg., medio para 4.43 seg. y mínimo para 5.43 seg.

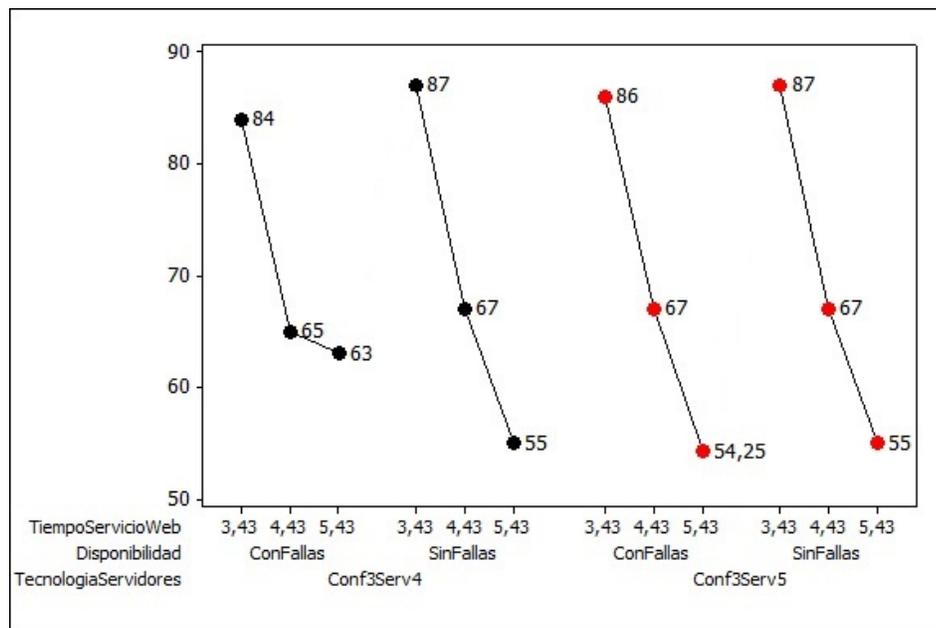


Figura 4.31: Cantidad de productos vendidos por día

### Análisis del Indicador Económico

En la Tabla 4.9 se calcula un costo total de inversión de U\$D 23.600 para un cluster con cinco servidores.

Con la consideración de los tres tiempos de servicios Web, la posibilidad de fallas y cinco servidores se calculan los ingresos por ventas mensuales (Tabla 4.16).

Si se decide acordar un tiempo de servicio Web de 3.43 seg. se mejora el ingreso por ventas en un 59 % respecto a 5.43 seg. y 28 % respecto a 4.43 seg.

Para completar el análisis se deben obtener las ganancias netas. En la Tabla 4.17 se

Productos Vendidos (día)	Productos Vendidos (mes)	Ingreso por Ventas (U\$D)
54	1620	48600
67	2010	60300
86	2580	77400

Tabla 4.16: Ganancias Netas para los diferentes tiempos de servicio Web

presentan las ganancias netas de acuerdo a los tres tiempos de servicios Web con fallas y cinco servidores, con un precio de venta de U\$D 30 por libro.

Tiempo de Servicio (seg.)	Productos Vendidos (día)	Porcentaje de Beneficio	Ganancia Neta Mensual (U\$D)
5.43	54	3.10 %	2244,67
4.43	67	4.07 %	2454,21
3.43	86	5.86 %	2847,96

Tabla 4.17: Ganancias Netas para los diferentes tiempos de servicio Web

Con los datos de las ganancias netas se calculan los meses que se necesitan para el retorno de la inversión y se los detallan en la Tabla 4.18.

Productos Vendidos (día)	Meses
54	11
67	10
86	8

Tabla 4.18: Meses para el retorno de la inversión

Por tanto, del análisis integrado de los indicadores tecnológicos y de negocios se sugiere diseñar un cluster de cinco servidores con las características del escenario 3 (procesador AMD Phenom 2 con 4 núcleos, memoria RAM DDR 3 de 4 GB y dos discos rígidos de 320 GB Raid 1) y contratar un servicio que asegure 3.43 seg. de tiempo de servicio Web.

Con esta decisión de negocios se alcanza el retorno de la inversión a partir del primer año de su implementación.

### 4.13. Conclusiones

Como primer paso para la demostración y uso de la metodología desarrollada se trabaja con datos tomados de un caso de estudio bibliográfico. Siguiendo el trabajo de los autores Menascé y otros (2000), se elabora un modelo de proceso de comercio electrónico tipo B2C para la venta minorista de libros.

La aplicación de la metodología no se centra en el aspecto de la plataforma informática necesaria para la implementación del proceso, sino más bien se sustenta en la estrategia de negocios planteada y los objetivos necesarios para lograr la misma.

Por ello se decide elaborar un modelo DEVS del proceso de comercio electrónico para la venta de libros con una estrategia totalmente virtual, de modo que el cliente puede concretar la compra del producto por medios electrónicos.

Para llevar adelante los objetivos se diseña un modelo de negocios que consiste en proponer los participantes y sus roles, plantear redes e identificar las transacciones comerciales.

Con base en el modelo de negocios se construye el proceso de comercio electrónico con la composición de las interacciones de servicios. Para ello se plantea una coreografía de interacciones con socios de negocios y una orquestación de transacciones internas. La coreografía se esquematiza en un diagrama de secuencia y la orquestación en un gráfico CBMG.

Se toma un CBMG del caso bibliográfico cuyas transacciones consisten en visitar la página principal del sitio, realizar búsquedas de un producto, navegar en el sitio, seleccionar un producto, agregarlo al carro de compras y por último la ejecución del

pago. El modelo se dice que es de clientes frecuentes porque se considera el caso para clientes habituales con un mayor porcentaje de compra.

La implementación del cobro electrónico es a través de un servicio tercerizado para lo cual se implementa una comunicación con Paypal para que éste ejecute el cobro mediante tarjeta de crédito.

Finalizado el desarrollo del nivel de negocios se plantea el modelo de recursos informáticos. En la búsqueda de disponibilidad y buenos tiempos de respuesta se decide elaborar una plataforma informática basada en cluster de servidores paralelos con posibilidad de fallas.

En bajo nivel se parte de un modelo de red de colas compuesto por un centro de cola que representa a un balanceador de carga y un grupo de  $n$  centros de demora que identifican a los servidores. El balanceador de carga opera las colas de requerimientos arribados mediante su asignación en forma secuencial a los servidores desocupados. Dichos servidores no administran colas.

Como complemento al modelo de recursos se incorpora la posibilidad de existencia de fallas, de modo que frente a una, el servidor queda inactivo por un tiempo determinado, luego se reinicia y puede recuperar el requerimiento en falla. No se considera la falla del balanceador de carga.

Completado el modelado del nivel tecnológico se deciden las métricas de salida. Para ello se cuenta con el Framework de Métricas de Desempeño de Modelos de Negocios Electrónicos propuesto en la Sección 2.6, del cual se consideran las métricas del tiempo de respuesta, velocidad de procesamiento, cantidad de requerimientos en cola por recurso, cantidad de productos vendidos y punto de equilibrio.

El paso siguiente es la construcción del modelo DEVS del proceso de comercio electrónico. Para ello se propone una técnica de elaboración de algoritmos que desde una abstracción inicial en un diagrama de estructura que identifique los componentes atómi-

cos y acoplados, los explote en un modelo base y finalmente especifique el formalismo.

Para la especificación del formalismo se elaboran diagramas Statechart a partir de los cuales se identifican los puertos de entrada y salida de cada componente atómico, los estados y las transiciones efecto de las funciones de transición internas y externas. Dichos diagramas son base para la elaboración del formalismo DEVS.

La implementación del modelo DEVS se realiza en la herramienta DEVSSJAVA, a su vez que se organiza la experimentación para la obtención de los parámetros del sistema.

Estos parámetros son los correspondientes al tiempo entre arribos y los tiempos de servicio de los recursos. Para la distribución del tiempo entre arribos se toma la propuesta por el caso bibliográfico y para los tiempos de servicio se utiliza la herramienta SoapUI.

Se instala la herramienta de monitorización en las computadoras de los tres escenarios y se realizan las experimentaciones para cada transacción. Es importante destacar que la medición de las transacciones internas (CBMG) se hace para el servidor corriendo en forma local, mientras que para el caso del cobro electrónico se ejecuta un servicio disponible vía Web.

En el proceso de verificación se ejecuta en paralelo a la programación del modelo. Esto es cada vez que se incorpora un modelo atómico o un acoplamiento se ejecutan casos de prueba y se siguen su ejecución a través del paso de mensajes. Para el modelo completo se toman los archivos de Log y se recalculan las métricas para comprar con las obtenidas por simulación.

Al iniciar la simulación nos encontramos en un estado de incertidumbre frente a un gran número de posibles configuraciones que resuelven el problema. El problema aquí es que el costo de simulación es importante y por tanto se debe encontrar una forma de minimizar el mismo.

Por ello se aplica el algoritmo para la obtención de configuraciones eficientes que

a partir de una planificación de capacidad encuentre una plataforma adecuada que sea un balance entre las métricas tecnológicas y los costos de inversión. Para lograr este balance se cuenta con los indicadores de las métricas para comparación.

Con la configuración adecuada se propone un diseño experimental estadístico como estrategia de análisis de los efectos de los factores y sus interacciones sobre las variables de respuesta.

Un problema fundamental es el tiempo dedicado a la simulación. Para poder encontrar valores significativos de las métricas es importante analizar el tiempo en que las mismas muestran un comportamiento estable del sistema. La estabilidad se observa cuando el sistema logra medidas similares entre las sucesivas corridas.

Si el sistema está saturado va a presentar un continuo incremento de las métricas del tiempo de respuesta y no así para la velocidad de procesamiento ya que los clientes van a estar esperando su atención en el sistema.

Una vez encontrado el tiempo en el cual el sistema se muestra estable se debe estimar un tiempo mínimo de simulación.

Pero este tiempo en algunos casos se va a encontrar exigido por las métricas de negocios, ya que para estimar el número de clientes que compran se necesita simular un tiempo en el cual la mayor cantidad de clientes acceden al sitio.

Además, el tiempo de simulación depende de la cantidad de archivos de Log que se generen y de la periodicidad de captura de datos.

Por ello, se deben tener en cuenta para la estimación del tiempo de corrida de una experimentación (i) estabilidad de las salidas, (ii) rango representativo de los accesos de clientes y (iii) cantidad de información y frecuencia de captura en archivos de Log razonables.

Del diseño experimental estadístico se demuestra que los factores cantidad de servidores, disponibilidad y tiempo de servicio Web son significativos en las variables de

respuesta tiempo de respuesta y velocidad de procesamiento.

De los resultados de simulación se concluye que es conveniente incorporar cinco servidores al cluster con las características del escenario 3 y acordar con el socio de negocios Paypal un tiempo en servicio Web de 3.43 seg. De este modo se aseguran tiempos de respuesta *Sobresalientes* a pesar de la posibilidad de fallas para el escenario planteado y se alcanzan retornos de la inversión a partir del primer año de su implementación.



# CAPÍTULO 5

## Caso de Estudio

*En este capítulo se implementa la metodología DEVS de modelización para simulación de transacciones de comercio electrónico en un caso de estudio real de una empresa de venta minorista de electrodomésticos. Con herramientas de monitorización en el sitio Web se obtiene la información necesaria para parametrizar el modelo CBMG y la carga de trabajo del sistema y los tiempos de servicio de los recursos informáticos. Se construye el modelo DEVS siguiendo la técnica propuesta en los capítulos anteriores, se realizan las simulaciones y se analizan los resultados. Se presenta además una variante del modelo de la plataforma tecnológica con una configuración de servidores en capas.*

### 5.1. Descripción del Problema

En el caso de estudio se aplica la metodología propuesta en una experiencia real de comercio electrónico tipo B2C para una empresa dedicada a la venta minorista de electrodomésticos, electrónica, computación, artículos para el hogar, accesorios, música y películas.

La empresa con casa central en la ciudad de Santa Fe, tiene sucursales distribuidas

en diferentes ciudades de todas las provincias, las cuales comercializan sus artículos del modo tradicional y cuenta con un sitio de comercio electrónico que tiene como principal fin el marketing. A través del sitio se promocionan productos, se brinda información, se ofrece la compra telefónica y se provee de servicios a los clientes durante la compra y posterior a la misma.

La modalidad de compra no es en línea, el cliente selecciona un producto y envía una solicitud electrónica, la empresa la recibe y a través de una comunicación telefónica concreta la operación de venta. La empresa no posee socios de negocios que estén involucrados en las operaciones del sitio.

## 5.2. Modelo de Negocios

La estrategia de negocios consiste en la propuesta de una cartelera publicitaria con un catálogo de productos, servicios para la interacción con el cliente y ejecución de ventas minoristas en forma telefónica. El tipo de negocios se enmarca en el business to consumer (B2C) entre la empresa y el cliente.

A continuación se elabora un modelo de simulación DEVS de un proceso de comercio electrónico B2C para evaluar el desempeño tecnológico en relación con los beneficios y el retorno de la inversión.

Para lograr el objetivo se plantean las siguientes metas específicas:

- Analizar el sitio de negocios existente y encontrar las transacciones fundamentales en el procesos de comercio electrónico, destacando la lógica seguida para el ingreso de dinero con ventas.
- Diseñar un gráfico CBMG para analizar el comportamiento de clientes y establecer la carga de trabajo, con las posibles sesiones y las probabilidades de interacción

entre las transacciones.

- Definir la tecnología informática que ejecutará el sistema.
- Simular el proceso de comercio electrónico en relación al comportamiento de la plataforma tecnológica y los beneficios económicos.
- Calcular el tiempo para el retorno del capital invertido.

### 5.3. Análisis del Sitio de Negocios

De la exploración del sitio Web de la empresa se identifican las siguientes transacciones de negocios:

#### 1. *Inicio de sesión y solicitud de ayudas:*

- Registración de un nuevo usuario.
- Inicio de sesión de usuario registrado.
- Solicitud de ayuda: venta telefónica, preguntas frecuentes, ranking de productos, ranking de notas, servicios de instalación profesional, atención telefónica al clientes, presentación de formas de financiación con tarjeta de la empresa y garantía extendida, entre otros servicios.

#### 2. *Búsqueda de productos:*

- Por búsqueda directa se ingresa un criterio y se puede distinguir producto o contenido.

- Por búsqueda avanzada se ingresa la palabra de referencia y luego se deben completar datos de listas desplegables por Rubro, Sub-rubro, Tipo de producto, Rango de precio y Marca.
3. *Comparación de productos:* como resultado de una búsqueda se muestran los productos con la posibilidad de ser comparados.
  4. *Barra de Menú:*
    - Todos: contiene un directorio de productos por rubro y subrubro.
    - Rubros: al ser seleccionados muestran una lista de productos por subrubros.
    - Promociones.
    - Contenidos: notas, guías de compras (presenta características y funcionalidades de un producto para ayudar en la decisión de compras), preguntas frecuentes y glosario (explicación de términos para los artículos de hogar).
  5. *Descargar música:*
    - Gratuita, con puntos adquiridos por compras anteriores.
    - Sujetas al pago.
  6. *Ventana que permite seleccionar promociones, ofertas y productos destacados.*
  7. *Catálogo en línea:* catálogo digital de la versión impresa de productos promocionados, con la posibilidad de acceder a los productos y enviar solicitud de compra.
  8. *Interacción con el cliente:*
    - Suscripción a Newsletter.

- Encuesta a usuario.

9. *Asesoramiento:*

- Atención telefónica.
- Guía de compras.
- Preguntas frecuentes

10. *Información financiera:* propuestas de descuentos y cantidad de cuotas, discriminadas por Bancos.

11. *Notas:* leer notas y dejar comentarios.

12. *Tarjeta de la empresa:* obtener información requerida para obtener la tarjeta ofrecida por la empresa y de sus posibilidades de financiación.

13. *Servicios:* entrega express, cambio de productos, venta telefónica, envío a domicilio, entrega acordada, garantía extendida, asistencia informática, instalación profesional.

De las acciones identificadas se elabora un modelo de alto nivel del sitio, el cual se presenta en la Figura 5.1 un resumen con las diferentes transacciones.

Del modelo de alto nivel del sitio se toman las diferentes transacciones, se establecen las transiciones entre ellas y se elabora un gráfico CBMG (Figura 5.2).

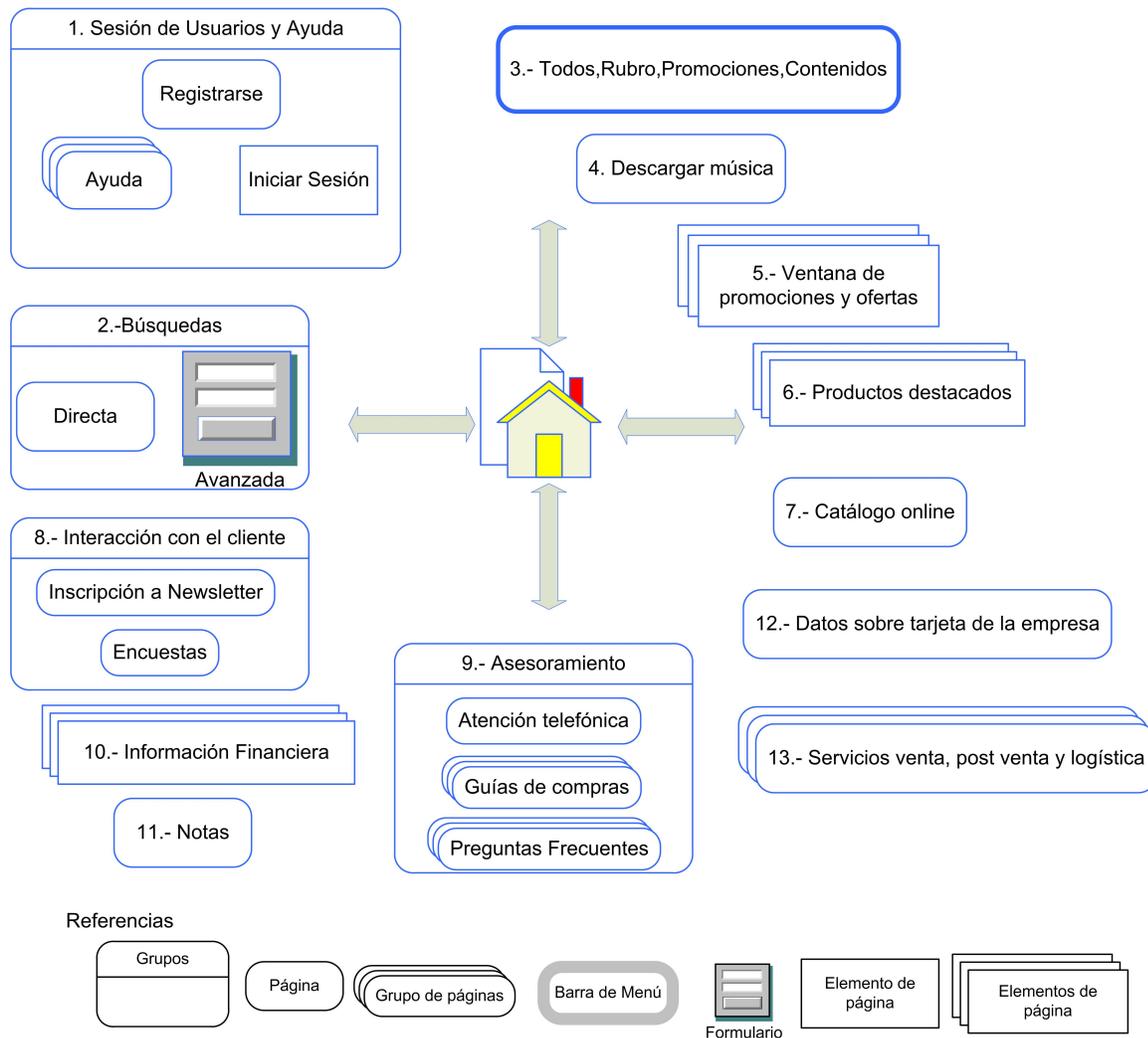


Figura 5.1: Modelo de alto nivel del sitio de negocios de la empresa

## 5.4. Propuesta de Transacciones y Servicios

Del gráfico CBMG de la Figura 5.2 se debe construir el que representa la sesión de clientes que se quiere modelar. Para ello se utiliza la información captada por el módulo e-commerce tracking de la herramienta en línea Google Analytics, cuya función consiste en monitorizar el sitio de negocios para seguir el comportamiento de los usuarios y medir las conversiones en ventas.

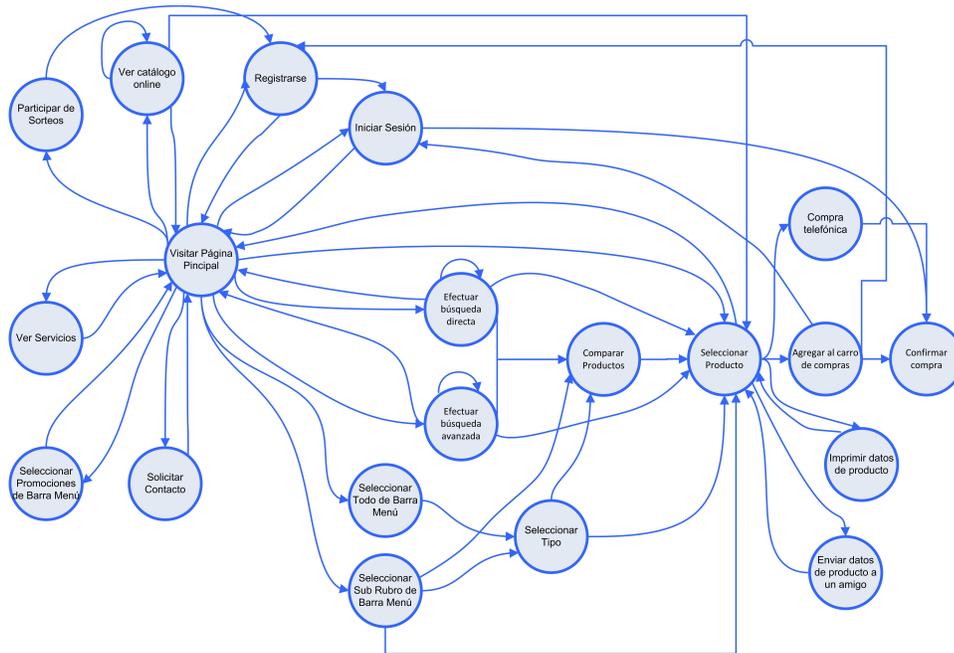


Figura 5.2: Gráfico CBMG del sitio

La información captada hace referencia a:

- Productos que el cliente compra, cantidad e ingresos generados por ventas.
- Transacciones: seguimiento del comportamiento de los clientes entre transacciones.
- Tiempo de compra: cantidad de días entre visitas y tiempos desde el inicio de una transacción hasta la partida.

Sobre la base de estos datos se conoce el nivel de aceptación de los productos por parte de los clientes, cuáles de ellos requieren mayor énfasis en marketing, los ingresos por ventas, los caminos seguidos por el cliente que concreta una compra, cuánto tiempo le toma al cliente decidirse a comprar, entre otras.

Se ejecuta la herramienta a través de un navegador de Internet en el mes de diciembre de 2011 y se determina la sesión de clientes de interés. En la figura 5.3 se presenta la

ventana en el punto inicial del recorrido de sesiones de clientes. En este caso los tres medios más importantes de ingreso son página inicial, catálogo y productos.

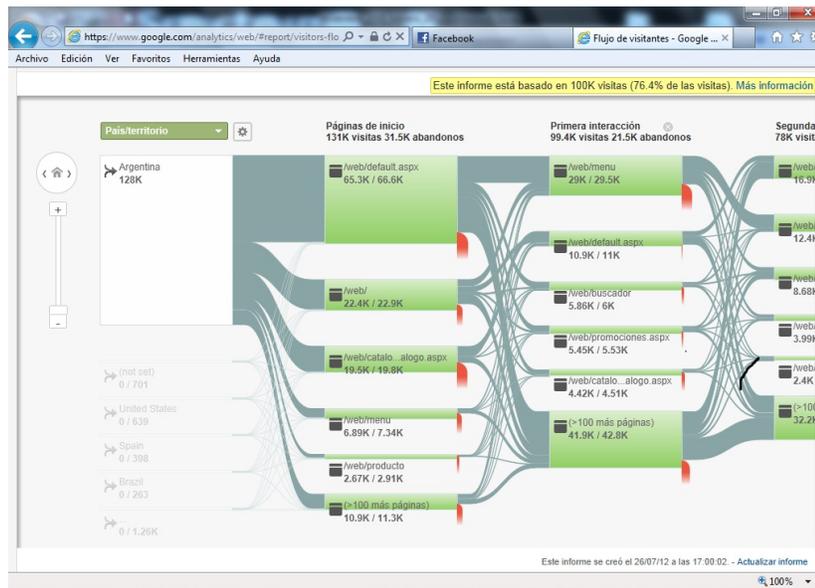


Figura 5.3: Primera ventana en el recorrido de sesiones de clientes

Para continuar con el seguimiento del comportamiento de clientes se analizan las diferentes interacciones. En la Figura 5.4 se muestra una ventana con la primera, segunda y tercera interacción.

Además del recorrido se puede determinar el porcentaje de clientes que siguen por las diferentes posibilidades de transición entre transacciones. La Figura 5.5 muestra la ventana en donde se observan los porcentajes, que son la fuente de datos para el cálculo de las probabilidades de transición de estados.

Para construir el modelo de sesiones se seleccionan las transacciones del sitio que son de interés en la modelización. En este caso se consideran las transacciones que siguen los clientes que compran y consisten en ingresar a la página principal, seleccionar opciones del menú, realizar búsquedas directas, visitar el catálogo, ver promociones, seleccionar un producto y enviar un requerimiento de compra.

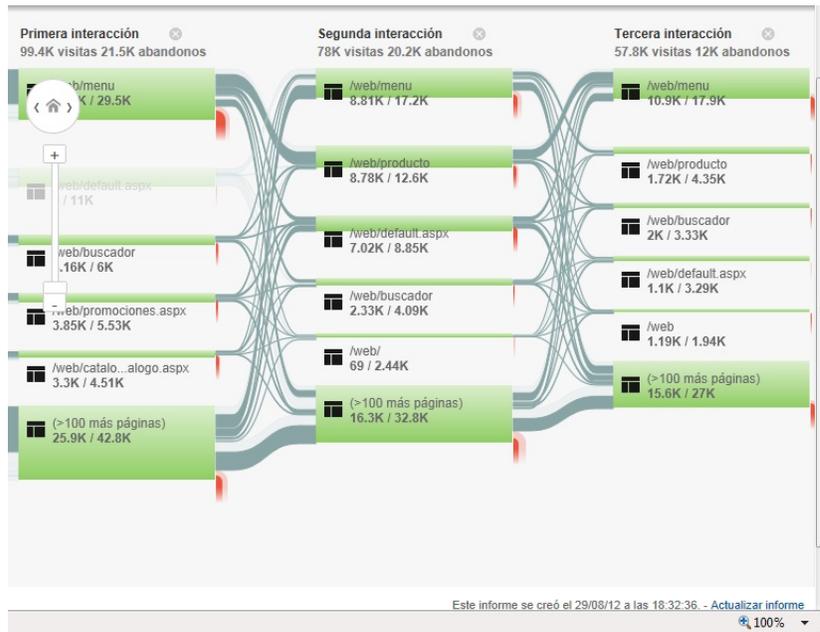


Figura 5.4: Ventana con la primera, segunda y tercera interacción

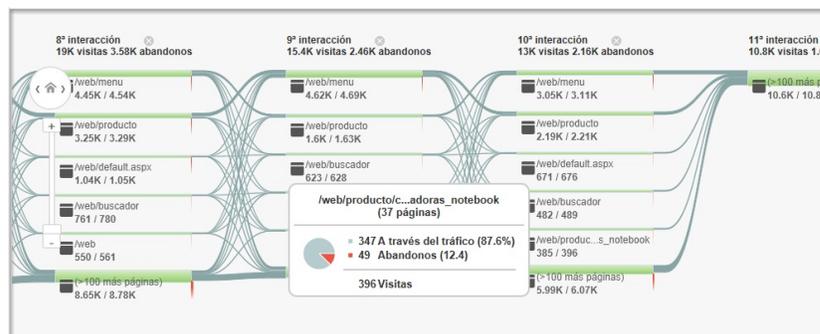


Figura 5.5: Ventana con porcentajes de transición de estados

## 5.5. Orquestación de Transacciones

Para la orquestación de transacciones se requiere la elaboración de un diagrama CBMG. En base a las transacciones obtenidas en la Sección 5.4 y sus transiciones se obtienen las probabilidades en el sistema de monitorización en línea Google Analytics. En la Figura 5.6 se muestra el diagrama CBMG obtenido.

En el mismo se proponen las diferentes transacciones de comercio electrónico y sus

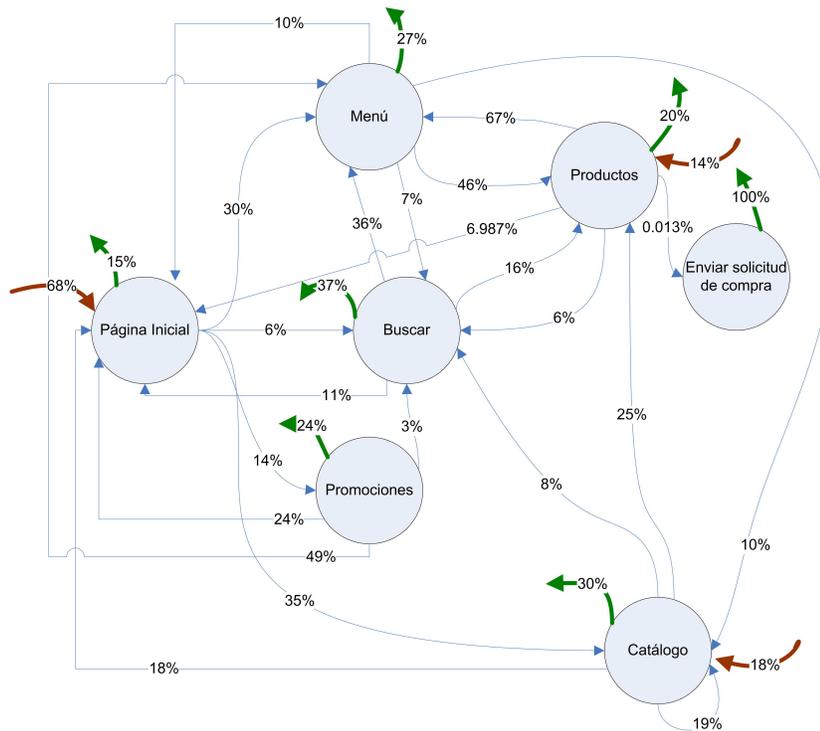


Figura 5.6: Diagrama CBMG de la sesión de cliente en estudio

probabilidades de transición. El ingreso al sitio se puede producir por las transacciones Página Inicial, Producto y Catálogo; mientras que la salida puede darse en cualquiera de ellas.

Un 4% de los clientes que selecciona un producto envía la solicitud de compra, la cual es respondida telefónicamente por la empresa para concretar la operación.

## 5.6. Modelo de Recursos

La plataforma informática del sitio está compuesta por:

- Servidor Win 2003 Standard 64 bits.
- LAN a 1 Giga por cobre.
- Firewall Cisco ASA5510.

Para la representación del modelo de recursos se utiliza el planteado en la Sección 4.4. En el mismo se planteaba un cluster de  $n$  servidores. Para este caso se parte del análisis con un único servidor y luego se definen nuevos escenarios con el incremento en su número.

## 5.7. Planteo de Métricas

### 5.7.1. Métricas a Nivel Tecnológico

- **Velocidad de Procesamiento:** número de requerimientos de clientes completados por unidad de tiempo.
- **Tiempo de Respuesta:** tiempo promedio de sesión de cliente en el sitio.
- **Utilización:** porcentaje de ocupación de los recursos.

### 5.7.2. 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 del punto de equilibrio:

$$\text{Costo Inversión} = \text{Cantidad de Productos Vendidos} \times \text{Ingreso Neto Promedio}$$

Según las cantidades de productos vendidos en promedio por mes se estima un ingreso promedio mensual. Se compara el Costo de Inversión con el Ingreso Neto mensual y se analiza el punto de equilibrio. Un valor menor a este punto indica pérdidas y un valor mayor, ganancias. Es importante tener en cuenta el tiempo en el cual se va a obtener este retorno.

## 5.8. Construcción del Modelo DEVS

Se comienza la construcción del modelo DEVS utilizando el diagrama de estructura de acuerdo a una metodología top-down (Figura 5.7). Como subestructuras principales se plantean el Marco Experimental (EF), el Modelo de Transacciones (TransM) y el Modelo de Plataforma Tecnológica (TechM). Para cada una de estas subestructuras se definen los componentes atómicos. El Marco Experimental consta de un Generador y un Transductor, mientras que el Modelo de Transacciones se compone de instancias de las diferentes transacciones en el sitio de negocios. La Plataforma Tecnológica, para esta propuesta, consiste en un cluster de servidores compuesto por un coordinador y servidores.

En el diagrama se pueden identificar los acoplamientos. Por tanto y en otro sentido, con una metodología bottom-up, desde los nodos hojas se identifica cada modelo atómico y en el nodo padre se escriben sus acoplamientos.

Entre cada nivel padre-hijo se agrega otro para indicar los acoplamientos entre puertos. El nivel ef-transm-techm-dec muestra los puertos y acoplamientos entre modelos principales. En los casos ef-dec, transm-dec y techm-dec se muestran los acoplamientos en cada subestructura.

Con el diagrama de estructura se construye el modelo base en la Figura 5.8. La misma comprende los tres modelos acoplados Experimental Framework, Transactional Model and Technological Model, unidos entre sí para formar el E-Commerce Model.

Respecto al Transactional Model se observa la existencia de un círculo con la letra P en su interior que deja expresado el comportamiento estocástico de la salida, es decir el puerto de salida se determina según una probabilidad.

El modelo tecnológico consiste en una plataforma basada en un cluster de servidores.

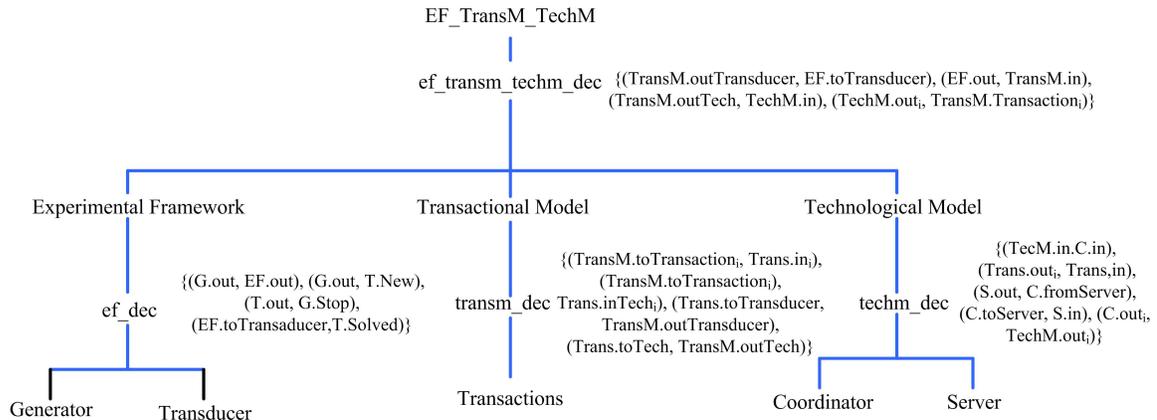


Figura 5.7: Diagrama de estructura del modelo de simulación de comercio electrónico

Un componente atómico Coordinator funciona como balanceador de cargas, gestionando la recepción de los requerimientos arribados con una asignación secuencial en los servidores. El balanceador de carga es responsable de mantener en cola los arribos que no pueden ser atendidos, por tanto los servidores administran colas.

## 5.9. Implementación en DEVSJAVA del Modelo de Comercio Electrónico

Como se mencionó, los parámetros de simulación se obtienen de los datos de monitorización con la herramienta Google Analytics. Sobre la base de dicha herramienta se decide trabajar con los datos del mes de diciembre del año 2011, ya que en el mismo se dispone de información del comportamiento en fechas comerciales con ventas picos como lo es Navidad.

Respecto al tiempo entre arribos se cuenta con un informe por mes en donde se conocen el número de clientes que arriban por día. En función de dicho informe se dispone de treinta y un datos de cantidades de clientes, a partir de los cuales se aplica un test de bondad de ajuste y se estima la distribución de probabilidad que mejor los

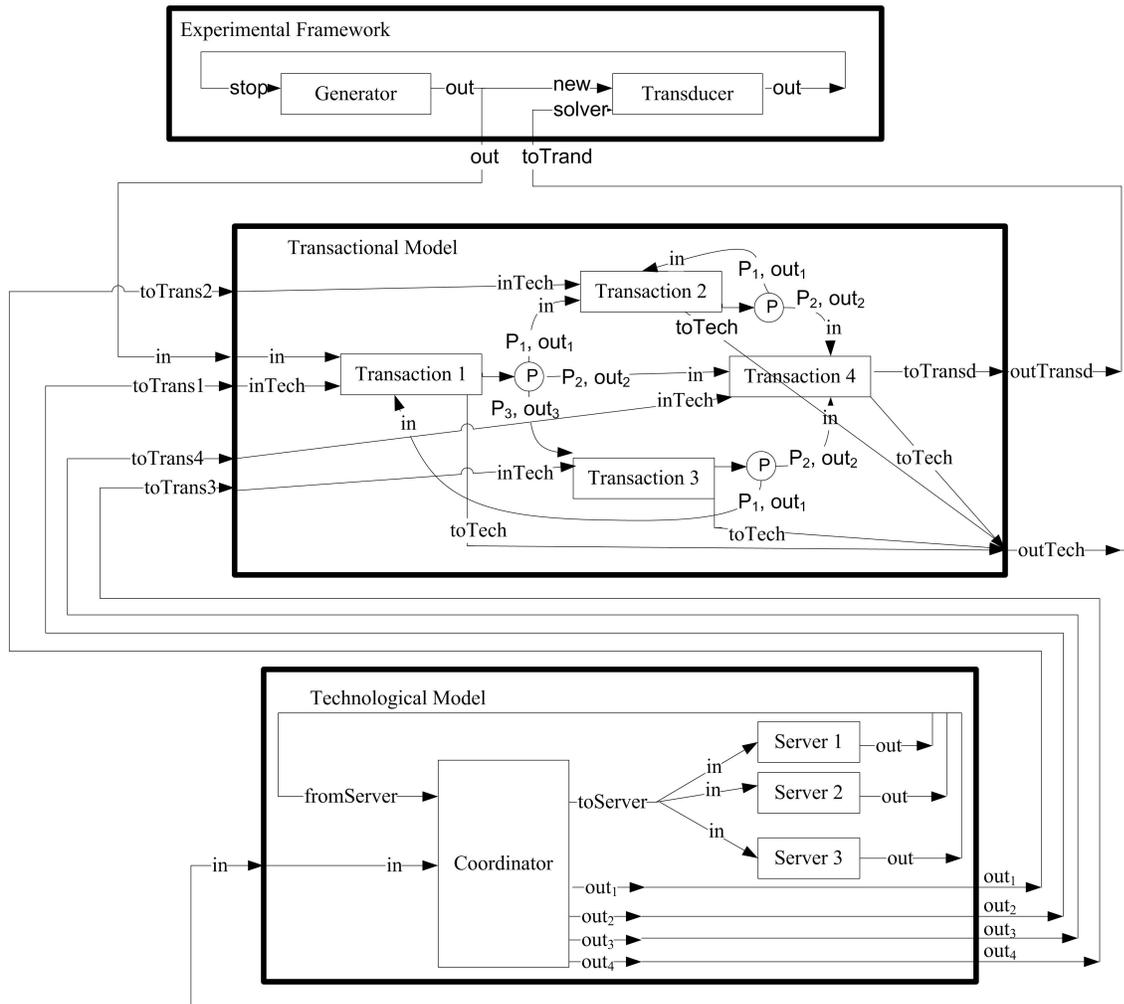


Figura 5.8: Modelo de Comercio Electrónico Base

representa.

De la aplicación del test de bondad de ajuste Kolgomorov-Smirnov en Minitab se estiman dos distribuciones posibles, una Exponencial con una media de 21.89 seg. de tiempo entre arribos de clientes, con un valor  $p$  0.003 y otra distribución Normal con una media de 21.89 seg. de tiempo entre arribos de clientes y desviación estándar 6.71 seg., con un valor  $p$  0.005.

Se experimenta con simulaciones para las dos distribuciones de probabilidad y se observa que la que mejor se aproxima a los datos reales es la Exponencial. Por ello se

trabaja con un modelo de carga en el generador con una distribución exponencial con una media de 21.89 seg. de tiempo entre arribos de clientes.

Para obtener los parámetros de la plataforma informática se utiliza el monitor del sistema operativo del servidor y se obtiene un tiempo de servicio promedio de transacción. Se estima una distribución uniforme con un mínimo de 0.5 seg. y el valor máximo de 1,5 seg. del tiempo de servicio del servidor.

En la Figura 5.9 se presenta el modelo DEVS implementado en la herramienta DEVJSJAVA.

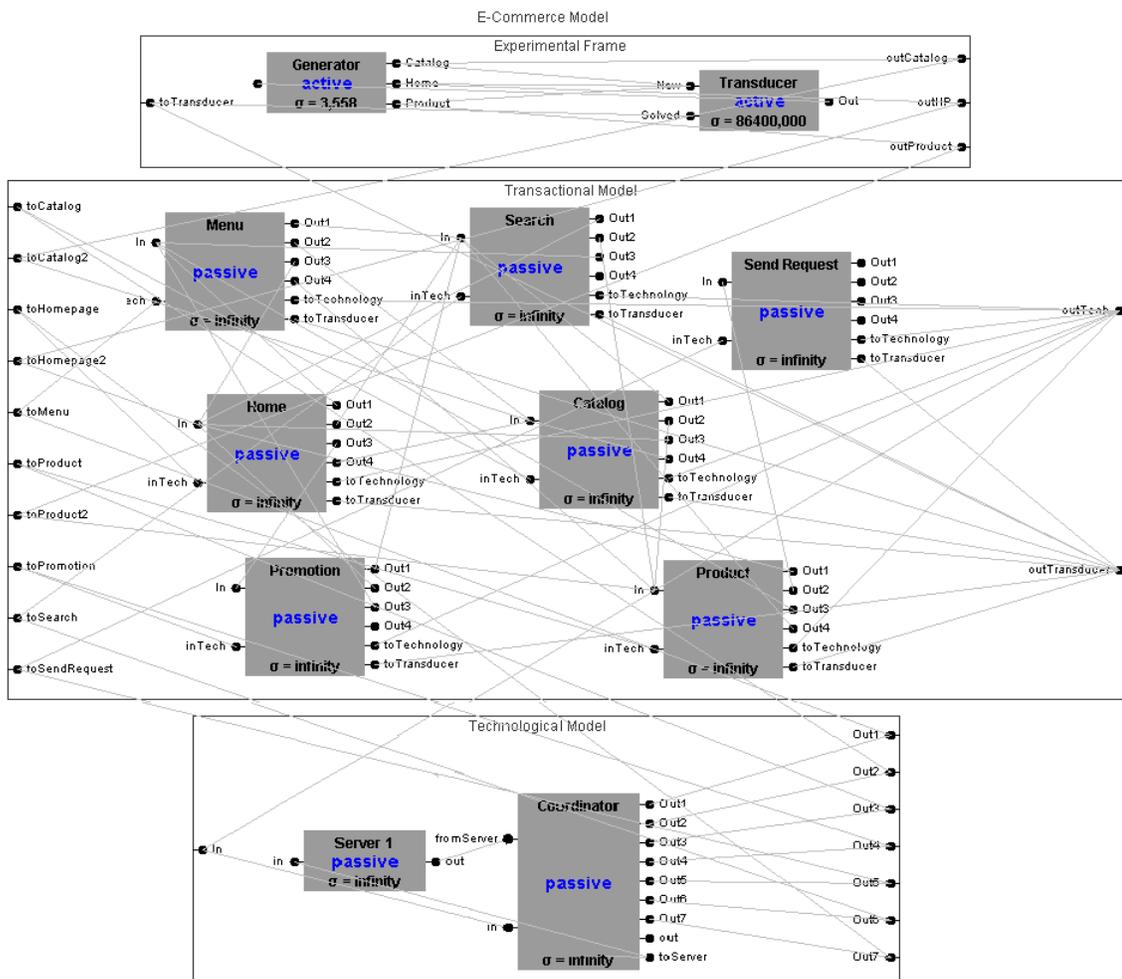


Figura 5.9: Modelo DEVSJAVA de la Empresa

En la plataforma tecnológica se propone un modelo de cluster de servidores en paralelo, organizado en un centro de cola que representa al balanceador de cargas. El mismo gestiona la recepción de los requerimientos arribados con una asignación secuencial en los servidores.

El balanceador de carga es responsable de mantener en cola los arribos que no pueden ser atendidos, por tanto los servidores se modelan como centros de demora.

En el sistema en estudio se cuenta con un único servidor, con lo cual el cluster está representado por un servidor, pero cuando se analizan posibles mejoras es necesario considerar más de uno.

## 5.10. Verificación y Validación del Modelo

La técnica de verificación se basa en el seguimiento de la ejecución del código y el análisis de los archivos de Logs con variables y métricas de salida. La validación se realiza comparando los resultados de la simulación con los obtenidos en el sitio real de negocios.

Se simula para un tiempo de 24 horas. Se toman datos del mes de diciembre de 2011 del sitio de negocios y se calculan valores promedio por día. En la Tabla 5.1 se muestran los valores promedio de la cantidad de clientes que arriban por día, el tiempo de respuesta del sistema percibido por los clientes y la cantidad de clientes que compran por día, tanto para el sistema real como para la simulación.

Métricas	Sitio Real	Sitio Simulado	Desviación Estándar
Cantidad de Clientes	4033	3889	1003
Tiempo de Respuesta	5 seg.	5.39 seg.	0.698
Clientes que compran	4.98	5	1.05

Tabla 5.1: Datos comparativos caso real y simulado

De la tabla se concluye que los valores obtenidos por simulación se ajustan a los datos del sistema real, por tanto el modelo de comercio electrónico es válido.

## 5.11. Aplicación del Algoritmo de Obtención de Configuraciones Eficientes

### 5.11.1. Simulación del Modelo Original para Planificación de Capacidad

Como primer escenario de simulación se analiza el comportamiento del sistema para un incremento en el número de clientes que ingresan al sitio de negocios. Se mantiene la tecnología existente y se disminuye el tiempo entre arribos con el fin de incrementar el número de clientes que arriban al sitio.

La Figura 5.10 muestra el número de clientes que ingresan por día al sitio en relación a los diferentes tiempos entre arribos.

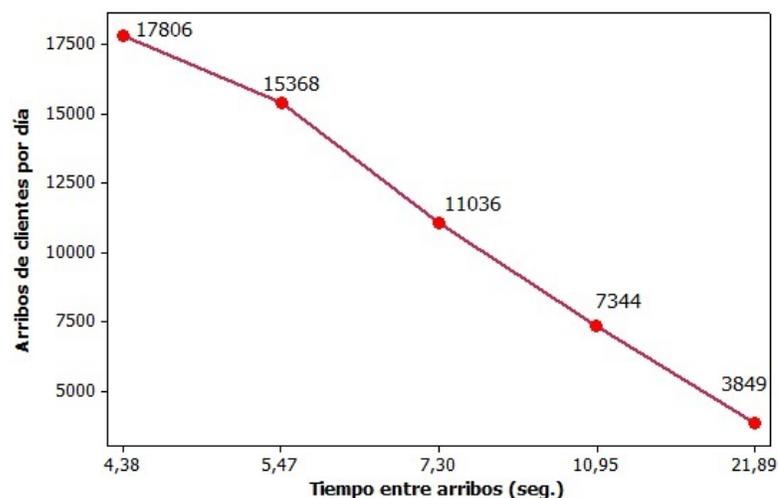


Figura 5.10: Número de clientes que arriban por día en función del tiempo entre arribos

En función del número de clientes que arriban por día se obtiene la cantidad de productos vendidos, lo que se muestra en Figura 5.11.

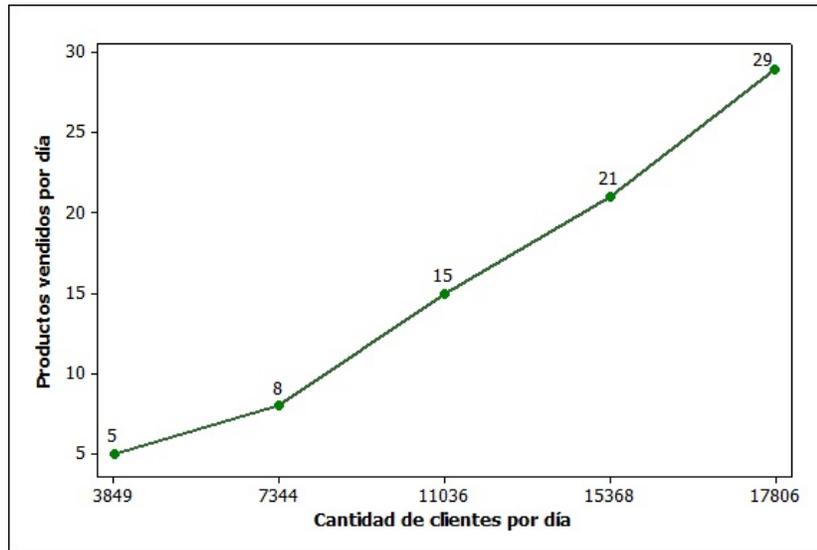


Figura 5.11: Cantidad de productos vendidos por día para los diferentes tiempos entre arribos

De la Figura 5.12 se advierte que al incrementar el número de clientes y disminuir el tiempo entre arribos, se alcanza un valor *Satisfactorio* del tiempo de respuesta de 8.94 seg. para 10.95 seg de tiempo entre arribos. Para el resto de los casos se obtienen valores *Insatisfactorios*.

Además para un tiempo entre arribos de 10.95 seg. se requieren 7344 clientes y se logra vender 8 artículos por día, con lo cual no es significativo en cuanto al beneficio económico.

En la Figura 5.13 se presentan las velocidades de procesamiento, en la cual se observa el crecimiento de la cantidad de requerimientos por segundo procesados por el incremento de arribos. Se evidencia un crecimiento de 4,5% entre el máximo y mínimo tiempo entre arribos.

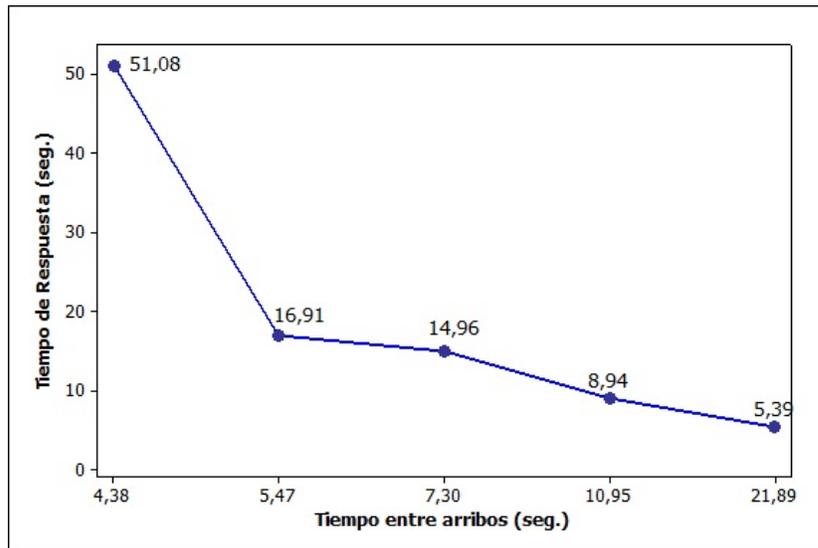


Figura 5.12: Tiempo de Respuesta Escenario 1

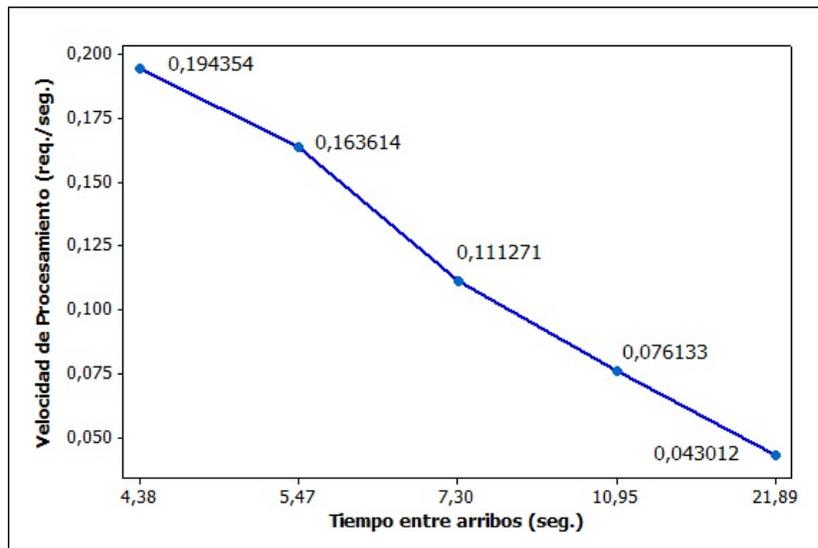


Figura 5.13: Velocidad de Procesamiento del Escenario 1

En la Tabla 5.2 se muestra la utilización del servidor para el Escenario 1. Para tiempos entre arribos de 4,38 seg., la utilización se encuentra próxima a la saturación, para 5,47 seg. y 7,30 seg. es alta y baja para el resto de los tiempos.

De las utilidades del servidor se concluye que frente a un crecimiento en el número de clientes que arriban, el sistema no respondería con un tiempo de respuesta adecuado.

Tiempo entre arribos (seg.)	Utilización
4,38	92 %
5,47	75 %
7,30	55 %
10,95	36 %
21,89	18 %

Tabla 5.2: Utilización del servidor del Escenario 1

Como segundo Escenario se propone trabajar sobre la tecnología con el fin de responder ante un incremento en el número de clientes. Para ello se incorpora un servidor al cluster de la misma característica del existente, formando otro de dos servidores paralelos.

El tercer Escenario consiste en cambiar la tecnología existente por un único servidor de mayor capacidad, con un tiempo de servicio correspondiente a una distribución de probabilidad Uniforme para un mínimo de 0.3 seg. y un máximo de 0.7 seg.

De la Figura 5.14 se concluye que es conveniente cambiar el servidor existente por uno de mayor capacidad ya que sus tiempos de respuestas son inferiores respecto a la propuesta de incorporar un servidor de la misma capacidad y formar un cluster.

En la Figura 5.15 se presentan las velocidades de procesamiento, de las cuales se analiza que el crecimiento de las mismas para el Escenario 3 es del 1.4 % sobre las del Escenario 2.

En la Tabla 5.3 se presentan las utilizaciones del servidor para los Escenarios 2 y 3, de las cuales se advierte que son aproximadamente iguales con una diferencia de 1 % para tiempos entre arribos de 21,89 seg. y 10,95 seg.

Del análisis de tiempo de respuesta y utilizaciones se concluye que es conveniente cambiar el servidor existente por uno que tenga las características del propuesto en el Escenario 3.

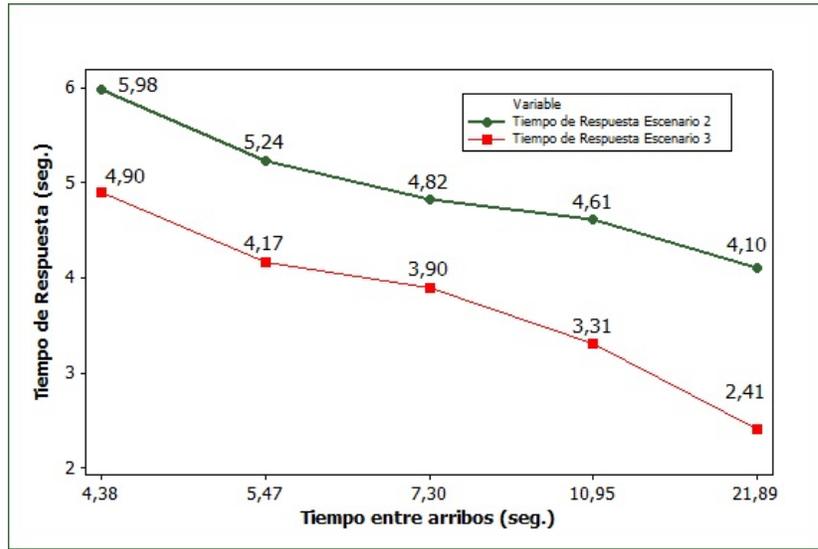


Figura 5.14: Tiempo de respuesta para los Escenarios 2 y 3

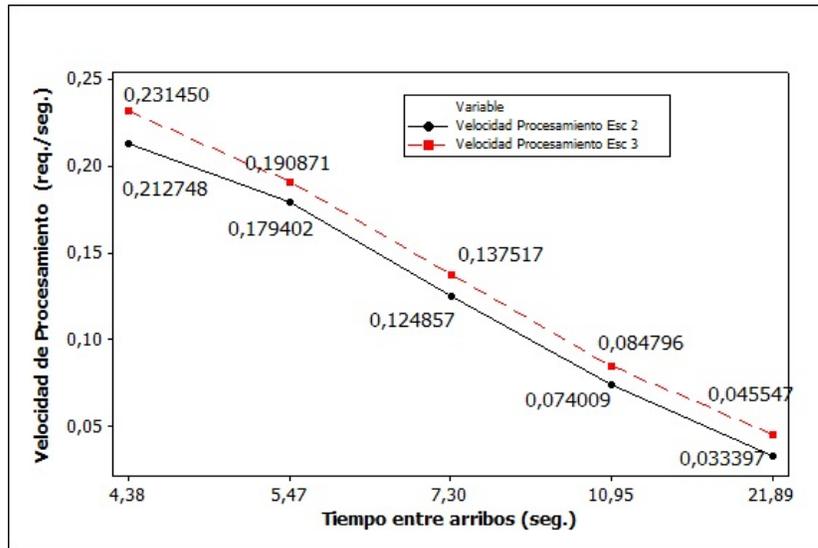


Figura 5.15: Velocidad de Procesamiento de los Escenarios 2 y 3

Por tratarse de variables aleatorias se completa el análisis con los valores de las desviaciones para el tiempo de respuesta. En las Tablas 5.4, 5.5 y 5.6 se presentan las desviaciones estándares para los tiempos de respuesta de los escenarios 1, 2 y 3 respectivamente.

Respecto a la desviación estándar de la velocidad de procesamiento se encuentra en

Tiempo entre arribos (seg.)	Utilización Esc. 2	Utilización Esc. 3
4,38	47 %	47 %
5,47	37 %	37 %
7,30	28 %	28 %
10,95	19 %	18 %
21,89	10 %	9 %

Tabla 5.3: Utilización del servidor para los Escenarios 2 y 3

Tiempo entre arribos (seg.)	Tiempo de Respuesta (seg.)	Desviación Estándar
21.89	5.39	1.24
10.95	8.94	1.05
7.3	14.96	1.23
5.47	16.91	1.45
4.38	51.08	1.74

Tabla 5.4: Desviación Estándar del Tiempo de Respuesta del Escenario 1

Tiempo entre arribos (seg.)	Tiempo de Respuesta (seg.)	Desviación Estándar
21.89	4.10	1.48
10.95	4.61	0.98
7.30	4.82	1.47
5.47	5.24	1.221
4.38	5.98	0.958

Tabla 5.5: Desviación Estándar del Tiempo de Respuesta del Escenario 2

Tiempo entre arribos (seg.)	Tiempo de Respuesta (seg.)	Desviación Estándar
21.89	2.41	0.988
10.95	3.31	1.43
7.30	3.90	0.924
5.47	4.17	0.991
4.38	4.90	1.42

Tabla 5.6: Desviación Estándar del Tiempo de Respuesta del Escenario 3

el rango entre 0.001 y 0.004 para los escenarios 1, 2 y 3.

### 5.11.2. Métrica de Negocios de los Escenarios 1, 2 y 3

Se debe obtener un punto de equilibrio entre el costo de la inversión y los beneficios netos. Para ello se calcula el costo de inversión para los diferentes escenarios como la suma de los costos de la plataforma informática y del desarrollo del sistema de comercio electrónico.

En la Tabla 5.7 se muestra un costo total de inversión para el Escenario 1. De modo que si se mantiene el sistema original el costo de inversión es de U\$D 115.190.

Concepto	Monto (U\$D)
Windows Compute Cluster Server 2003	590
Cisco ASA 5510 Adaptive Security Appliance	2.100
Costos del desarrollo del sitio de negocios electrónicos	112.500
<b>Costo Total de Inversión</b>	<b>115.190</b>

Tabla 5.7: Costos de Inversión Escenario 1

En la Tabla 5.8 se muestra un costo total de inversión de U\$D 116.000 para el Escenario 2, con la incorporación de dos servidores en cluster.

Concepto	Monto (U\$D)
Windows Compute Cluster Server 2003 con dos servidores	1.400
Cisco ASA 5510 Adaptive Security Appliance	2.100
Costos del desarrollo del sitio de negocios electrónicos	112.500
<b>Costo Total de Inversión</b>	<b>116.000</b>

Tabla 5.8: Costos de Inversión Escenario 2

En la Tabla 5.9 se muestra un costo total de inversión de U\$D 119.150 del Escenario 3, con el cambio por un nuevo servidor de mayor capacidad.

Se necesita saber cuál es la ganancia neta obtenida para comparar con el costo total de inversión y calcular en cuánto tiempo se podrá recuperar el mismo. Para ello se

Concepto	Monto (U\$D)
Windows Server 2012	4.550
Cisco ASA 5510 Adaptive Security Appliance	2.100
Costos del desarrollo del sitio de negocios electrónicos	112.500
<b>Costo Total de Inversión</b>	<b>119.150</b>

Tabla 5.9: Costos de Inversión Escenario 3

trabaja sobre los beneficios obtenidos de diferentes cantidades de clientes que arriban al sistema.

El cálculo de los beneficios se basa en los movimientos de ventas del mes de diciembre de 2011, provisto por Google Analytic. Del informe se manifiesta que los productos más vendidos son splits, notebooks y teléfonos celulares.

Considerando que en un tiempo entre arribos de 21,89 seg. se venden en promedio 5 artículos, en la Tabla 5.10 se presentan los artículos, sus precios unitarios, el porcentaje de venta de cada producto respecto al total y la incidencia de cada artículo por día.

Producto	Precio Unitario (U\$D)	Porcentaje de venta	Cantidad
Split	1.000	50 %	2.5
Notebook	1.250	30 %	1.5
Teléfono Celular	125	20 %	1

Tabla 5.10: Productos más vendidos por día

Estimando una venta promedio de 150 productos al mes, se calcula un ingreso por ventas de U\$D 675.000. Se necesita saber cuál es el ganancia neta obtenida. De acuerdo a la Ecuación 2.10:

$$Ganancia\ Neta = U\$D\ 675,000 * Porcentaje\ de\ Beneficio$$

De la aplicación de la Ecuación 2.11 con datos del Estado de Resultados de la empresa para diciembre de 2011, se obtiene un *Procentaje de Beneficio* de 2,58 %.

Por tanto la Ganancia Neta es:

$$\text{Ganancia Neta} = U\$D 675,000 * 2,58 \% = U\$D 17,415$$

En la Tabla 5.11 se detallan los datos sobre tiempos entre arribos de clientes, cantidad de clientes, productos vendidos y beneficios netos para los diferentes tiempos entre arribos.

Tiempo entre arribos (seg.)	Cantidad de clientes	Productos vendidos	Ganancia Neta (U\$D)
4,38	17806	29	101.007
5,47	15368	21	73.143
7,30	11036	17	59.211
10,95	7344	8	27.864
21,89	3849	5	17.415

Tabla 5.11: Resultado de negocios

Para dichos datos se muestran los meses necesarios para cubrir los costos de inversión para los tres escenarios, en la Tabla 5.12.

Tiempo entre arribos (seg.)	Meses Esc. 1	Meses Esc. 2	Meses Esc. 3
4,38	1,13	1,14	1,17
5,47	1,57	1,58	1,62
7,30	1,94	1,95	2,01
10,95	4,13	4,16	4,27
21,89	6,61	6,66	6,84

Tabla 5.12: Meses para retorno de inversión en Escenarios 1, 2 y 3

En conclusión si se selecciona la alternativa del Escenario tres, que se destacó como adecuada desde la perspectiva tecnológica se requieren siete meses para el retorno de la inversión.

Para disminuir el número de meses se requiere mayor número de clientes. De acuerdo al arribo de clientes actuales en el sitio se requieren dos años para cubrir la inversión.

A medida que se incrementa este número se disminuyen los meses necesarios.

De experiencias de comercio electrónico de empresas del mismo rubro se conoce que implementado el cobro electrónico en línea a través de tarjetas de crédito se logra incrementar el porcentaje de clientes que compran a un 3 %.

Es base a los parámetros del Escenario 1 se incrementa a 3 % la probabilidad de que los clientes que estando en la transacción *Product* pasan a enviar la solicitud de compra. Para un tiempo entre arribos de 21.89 seg. se observa que de 5 productos vendidos por día pasan a 77, con un tiempo de respuesta de 5.6 seg. y una utilización del servidor del 19 %.

Por lo cual se concluye que es conveniente evaluar una estrategia de negocios con la implementación de cobro electrónico.

## 5.12. Propuesta de Mejora con la Incorporación de Servicios Web

Como nueva estrategia se considera la alternativa de negocios totalmente virtual, es decir que el cliente tenga la posibilidad de hacer su pago por medios electrónicos. Por tanto, la empresa es responsable de diseñar una plataforma que ofrezca al cliente navegabilidad con indicadores de tiempo de respuesta adecuados y a su vez muestre confiabilidad para el cobro por medios electrónicos.

Como estrategia se decide que la empresa tenga a su cargo las transacciones del sitio para la navegación y búsqueda de productos, mientras que el cobro se articule por un servicio tercerizado.

Respecto a la gestión del inventario se decide coordinar con el proveedor para una operación integrada con el mismo. A nivel de sistema se determina que el cobro y la

administración de inventario se implementarán con tecnologías de servicios Web.

### 5.12.1. Coreografía de Servicios con Socio para Control de Inventario

En la Figura 5.16 se presenta la coreografía en el diagrama de secuencia que integra al sitio con el proveedor para el control de existencia en inventario.

Cada vez que un cliente confirma la compra de los productos en su carro de compras el sistema consulta el inventario a través de un servicio Web del proveedor. El sitio de ventas como consumidor de servicio envía un mensaje *Request* con los productos a verificar al proveedor y recibe a través de un mensaje *Response* la orden verificada.

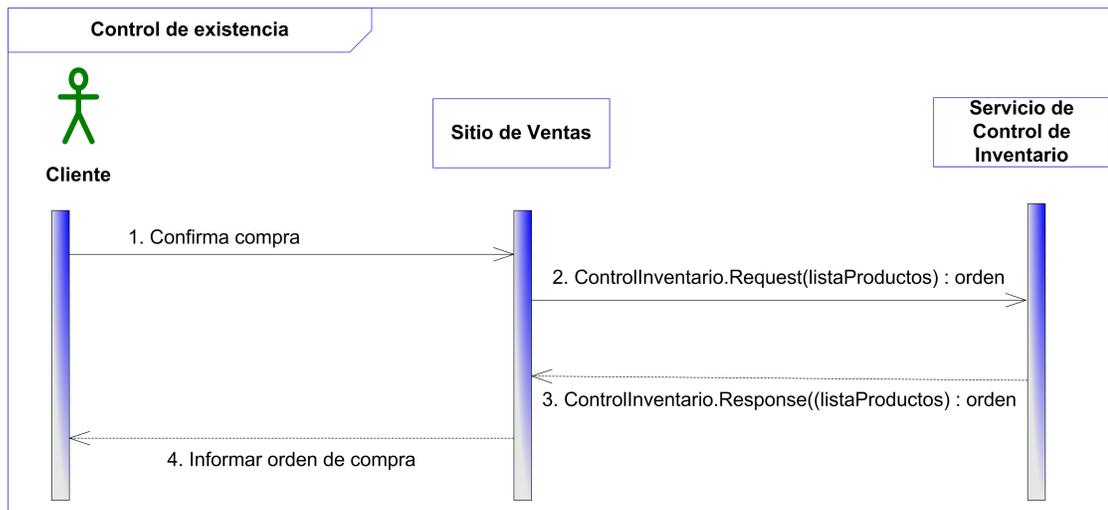


Figura 5.16: Diagrama de secuencia de integración con el proveedor

### 5.12.2. Coreografía de Servicios con Socio para Cobro Electrónico

Del mismo modo en la Figura 5.17 se presenta la coreografía de la integración con el socio para cobro electrónico, una vez que el cliente autoriza el pago el sitio envía un

mensaje *Request* al proveedor de servicio de cobro electrónico y recibe como respuesta la ejecución del mismo. Para este caso se considera el pago a clientes autorizados.



Figura 5.17: Diagrama de secuencia de integración con servicio de cobro

### 5.12.3. Orquestación Interna de Transacciones con Servicios Web

La Figura 5.18 muestra el diagrama de CBMG de la sesión en estudio, en el cual se incorpora el carro de compras, la consulta de existencia de stock al proveedor y el cobro a través de un medio electrónico.

Se presenta el nuevo modelo, de modo que con la confirmación de compra del cliente se verifica el inventario y se procede al cobro. Las probabilidades de transición del nuevo modelo se calculan en función de experiencias en el negocio de personal experimentado en el tema.

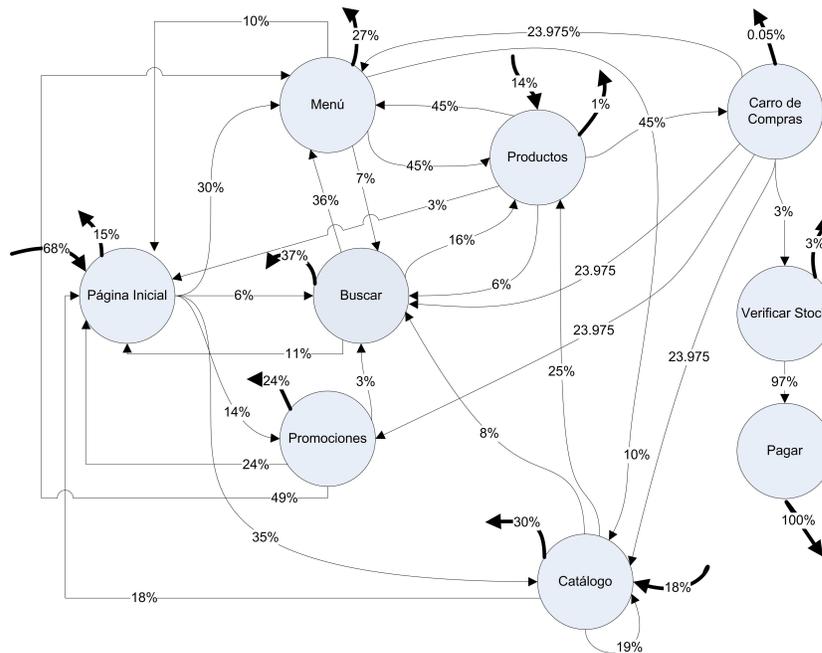


Figura 5.18: Diagrama CBMG de la sesión de cliente con servicios Web

### 5.13. Modelo DEVSJAVA con Servicios Web

Se replantea el modelo de simulación a nivel transaccional con la incorporación de los servicios Web. En la Figura 5.19 se muestra la implementación en DEVSJAVA.

Se observa el modelo transacciones en el cual se muestra la implementación del carro de compras en relación con el servicio Web del proveedor, pagar con su conexión al servicio Web de cobro y el resto de las transacciones con ejecución en la plataforma informática propia.

La obtención de las demandas de tiempos de servicios Web se realiza con el uso de la herramienta de software libre SoapUI (<http://www.soapui.org>), que posibilita el testeo de tecnologías con servicios Web. SoapUI cuenta con la función MockServices que simula un servidor de Web Services y obtiene los tiempo de respuesta ante un requerimiento de un consumidor.

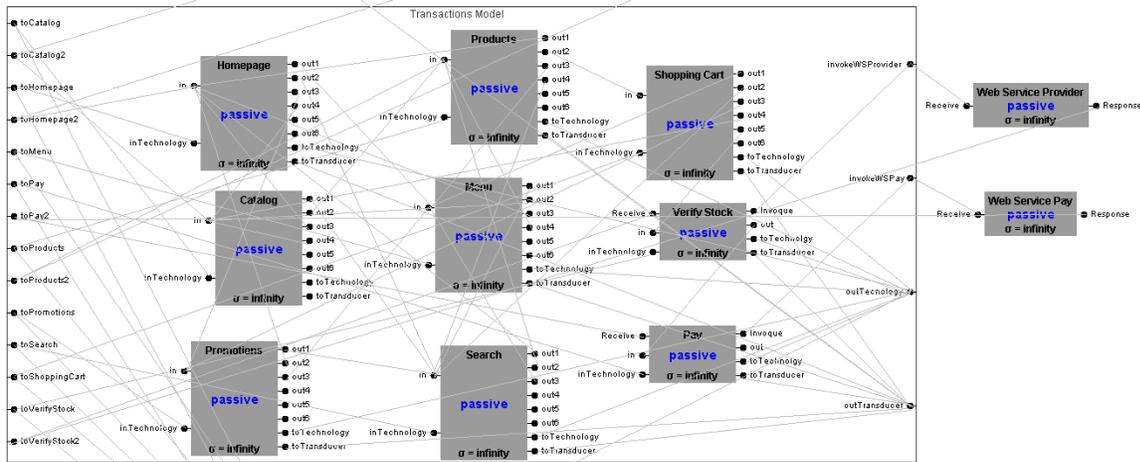


Figura 5.19: Modelo DEVSJAVA transaccional del nuevo CBMG

A partir de la experimentación se propone una distribución de probabilidad uniforme con mínimo 0.449 seg. y máximo 0.750 seg. para el tiempo asociado al control de inventario y una distribución de probabilidad uniforme con mínimo 0.849 seg. y máximo 0.990 seg. para el tiempo de la transacción pagar.

Este tiempo se conforma por la suma de los tiempos de envío del mensaje request, el servicio en la operación y la devolución del mensaje response.

Para el modelo de la plataforma informática de la nueva estrategia se tiene en cuenta un único servidor con una distribución de probabilidad Uniforme del tiempo de servicio para un mínimo de 0.5 seg. y un máximo de 1.5 seg y se consideran los parámetros originales para el resto de los recursos.

## 5.14. Simulación del Modelo con Servicios Web

Se plantea un Escenario 4 en el cual se incorporan los servicios Web de control de inventario y cobro electrónico. Se simula para diferentes tiempos entre arribos.

En la Figura 5.20 se presentan los tiempos de respuesta, en la cual se observa que

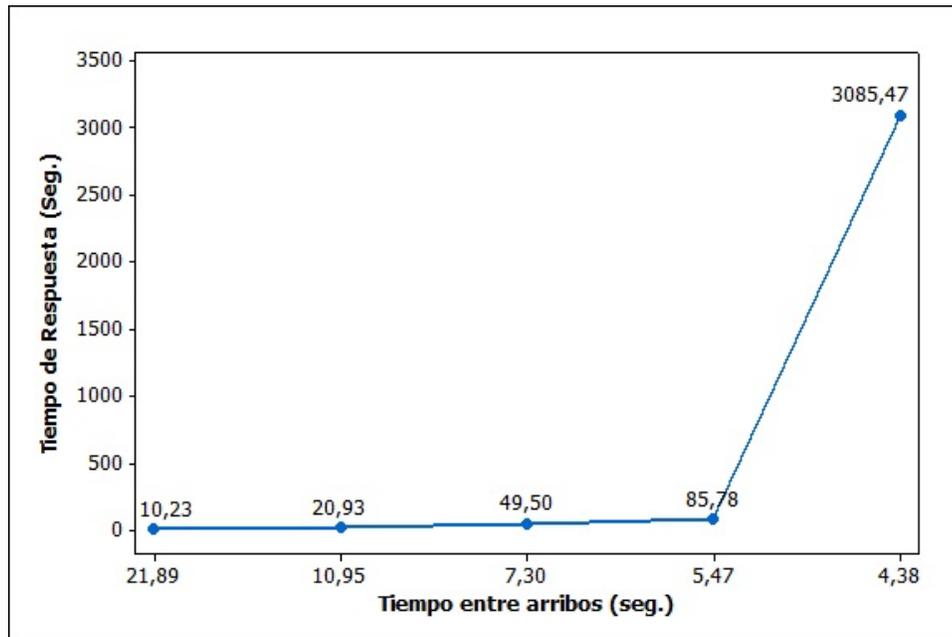


Figura 5.20: Tiempo de Respuesta para Escenario 4

para un tiempo entre arribos de 21.89 seg. el tiempo de respuesta se encuentra en el límite superior del nivel satisfactorio del indicador, ya que para valores superiores a 10.6 seg. es insatisfactorio. Para 10.95, 7.30 y 5.43 la métrica es insatisfactoria, con una saturación del sistema en 4.38 seg. del tiempo entre arribos.

En la Tabla 5.13 se presentan las desviaciones estándares para el tiempo de respuesta del Escenario 4.

Tiempo entre arribos (seg.)	Tiempo de Respuesta (seg.)	Desviación Estándar
21.89	10.23	0.184
10.95	20.93	1.31
7.30	49.50	2.46
5.47	87.58	3.74
4.38	3080.47	189.81

Tabla 5.13: Desviación Estándar del Tiempo de Respuesta del Escenario 4

En la Tabla 5.14 se muestra la utilización del servidor para el Escenario 4.

Tiempo entre arribos (seg.)	Utilización
4,38	100 %
5,47	94 %
7,30	70 %
10,95	46 %
21,89	25 %

Tabla 5.14: Utilización del servidor del Escenario 4

En función del número de clientes que arriban por día se muestra en la Figura 5.21 la cantidad de productos vendidos.

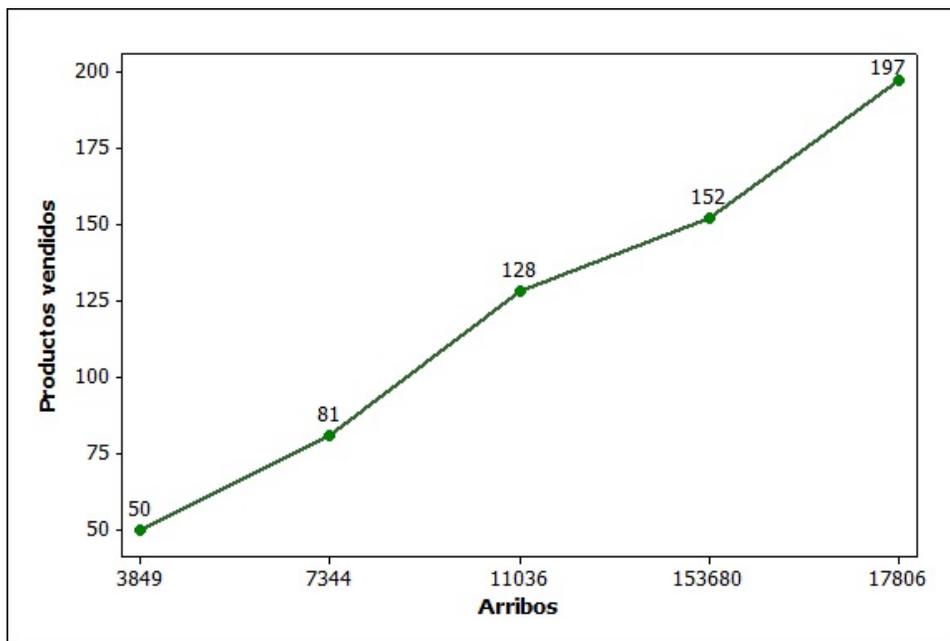


Figura 5.21: Cantidad de productos vendidos por día

Tomando como referencia el tiempo entre arribos de 21.89 seg y 3449 arribos al mes se venden 50 productos en promedio por día. El monto ingreso neto por mes de artículos vendidos es de U\$D 117.000, al cual se le sustrae el 3 % correspondientes al pago del servicio a la empresa de cobro electrónico, resultando un valor final de U\$D 113.490.

Como se mencionó en la Subsección 5.11.2, la inversión inicial en el desarrollo del sitio

es de U\$D 112.500, con un balance positivo de U\$D 990. De los resultados del Escenario 3 se conoce que el cambio de un servidor de mayor capacidad logra los indicadores de performance deseados.

Por ello se propone en el Escenario 5 una plataforma tecnológica compuesta por un servidor con una distribución de probabilidad Uniforme del tiempo entre arribos para un mínimo de 0.3 seg. y un máximo de 0.7 seg.

Los tiempos de respuesta se presentan en la Figura 5.22 y muestran que cumplen con los límites del indicador.

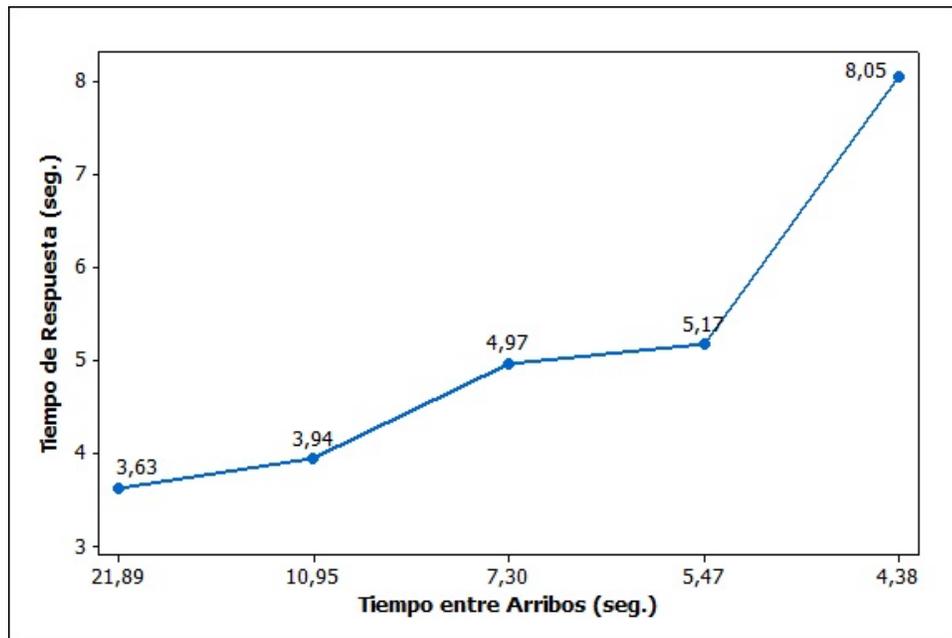


Figura 5.22: Tiempo de Respuesta del Escenario 5

En la Tabla 5.15 se presentan las desviaciones para el tiempo de respuesta del Escenario 5.

Como los tiempos de respuesta son satisfactorios y el balance económico es positivo se concluye que es suficiente con el nuevo servidor para soportar el proceso de comercio electrónico y a su vez obtener beneficios aceptables.

Tiempo entre arribos (seg.)	Tiempo de Respuesta (seg.)	Desviación Estándar
21.89	3.64	0.57
10.95	3.94	0.87
7.30	4.97	1.78
5.47	5.17	1.97
4.38	8.05	2.89

Tabla 5.15: Desviación Estándar del Tiempo de Respuesta del Escenario 5

El balance positivo obtenido puede ser utilizado como inversión en el equipamiento necesario.

Además de las métricas planteadas se puede trabajar a nivel de comportamiento de clientes en el Sitio. A través de un archivo de Log se registran los diferentes recorridos seguidos por los clientes en su navegación.

En la Tabla 5.16 se muestran algunos recorridos agrupados por su frecuencia.

Del análisis se puede determinar cuáles son las transacciones más visitadas por los clientes y de este modo incorporar facilidades para la compra y promoción de productos. El resto de los recorridos utiliza diferentes caminos con frecuencias entre 10 y 1.

Otra posibilidad de análisis es encontrar los caminos que siguen los clientes que compran. En la Tabla 5.17 se muestran los recorridos con frecuencia mayor que 1.

De los recorridos de la Tabla 5.17 se observa que los clientes que compran no realizan búsquedas de los productos sino que acceden a través del menú y del catálogo.

Del comportamiento de clientes se analiza que 1878 agregaron productos al carro de compras, 53 autorizaron la compra y 50 ejecutaron el pago. Es decir, el 2,6 % de clientes que agregan productos al carro de compras llegaron a efectuar el pago. Respecto al control de stock en el proveedor un 94,3 % de los clientes que autorizaron la compra tuvieron una respuesta positiva al control de inventario. Estos valores verifican el gráfico CBMG de transacciones de la Figura 5.18.

Recorrido	Frecuencia
Homepage	383
Catalog	348
Homepage, Catalog	278
Homepage, Menu	195
Homepage, Promotions	82
Homepage, Search	78
Homepage, Catalog, Catalog	64
Catalog, Catalog	56
Homepage, Menu, Products, Menu	41
Homepage, Catalog, Search	40
Homepage, Promotions, Menu	39
Catalog, Search	34
Homepage, Catalog, Products, Menu	33
Catalog, Homepage	29
Catalog, Products, Menu	29
Homepage, Search, Menu	24
Homepage, Menu, Catalog	24
Catalog, Homepage, Catalog	22

Tabla 5.16: Transacciones frecuentes

Recorrido	Frecuencia
Homepage, Menu, Products, Shopping Cart, Verify Stock, Pay	5
Homepage, Menu, Products, Menu, Products, Shopping Cart, Verify Stock, Pay	4
Catalog, Catalog, Products, Shopping Cart, Verify Stock, Pay	2
Homepage, Catalog, Products, Shopping Cart, Verify Stock, Pay	2

Tabla 5.17: Transacciones más frecuentes de clientes que compran

Además podemos calcular que 1528 clientes tuvieron intención de compra pero no concretaron la misma.

## 5.15. Diseño Experimental Estadístico

Se propone evaluar el rendimiento de la plataforma informática en relación con la productividad de negocios del proceso de comercio electrónico. Se trabaja con dos alternativas de diseño tecnológico, una que consiste en incrementar el número de servidores de la misma capacidad y otra en cambiar el sistema con la incorporación de servidores de mayor capacidad.

Para estas alternativas se calcula la cantidad de clientes que concretan la compra y de este modo se proyecta el rendimiento económico.

El objetivo del experimento es analizar el efecto de la plataforma informática en las variables de respuesta para la evaluación integrada de productividad tecnológica y de negocios.

Para lograr el objetivo se seleccionan los factores y sus niveles que se exhiben en la Tabla 5.18.

Factores	Niveles	Tiempo de Servicio (seg.)
Número de Servidores	1	0.5-1.5
	2	
	3	
Capacidad de los Servidores	Original	0.5 - 1.5
	Mejorar 50 %	0.25 - 0.75
	Mejorar 75 %	0.125 - 0.375

Tabla 5.18: Factores y niveles del experimento

Las variables de respuesta se detallan en la Tabla 5.19 y son el tiempo de respuesta, la velocidad de procesamiento y la utilización de los recursos informáticos en el nivel tecnológico y el número de clientes que compran en el nivel de negocios.

Se considera un diseño multifactorial completo, con modelo de efectos fijos, con el fin de estudiar el efecto de los factores y sus interacciones en las variables de respuesta.

Grupo	Variable de Respuesta
Tecnológicas	Tiempo de Respuesta Velocidad de Procesamiento Utilización Servidores
Negocios	Número de clientes que compran

Tabla 5.19: Variables de Respuesta

Por cada nivel se consideran cuatro repeticiones de cada experimentación.

Con el propósito de obtener la cantidad de productos vendidos por día se experimenta para 24 h. de tiempo de simulación.

### 5.15.1. Análisis de Salidas de la Variable de Respuesta Tiempo de Respuesta

Antes de realizar el análisis de resultados del experimento es necesario evaluar el cumplimiento de los supuestos del modelo, para lo cual se realiza la comprobación de idoneidad por medio de gráficos. El gráfico de comprobación de idoneidad del modelo para la variable de respuesta tiempo de respuesta se presenta en la Figura 5.23.

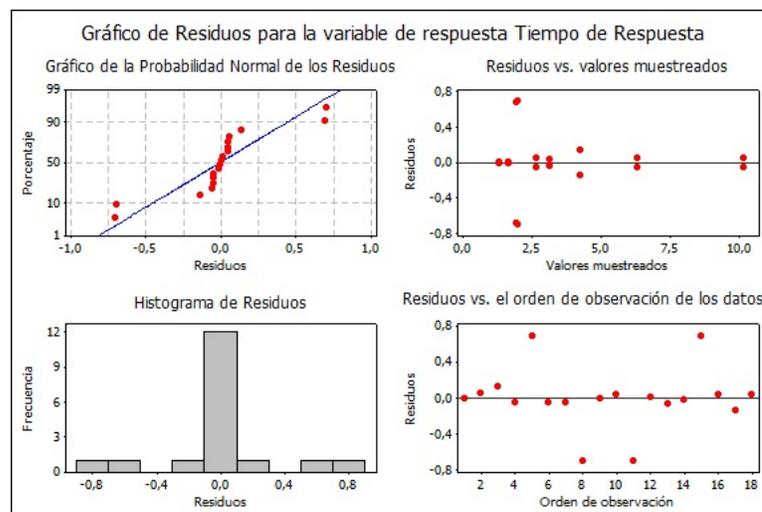


Figura 5.23: Comprobación de Idoneidad de la variable de respuesta Tiempo de Respuesta

Del gráfico de comprobación de idoneidad del modelo se concluye que se cumple la probabilidad normal de los residuos y la aleatoriedad de los datos muestreados.

En la Tabla 5.20 se detalla el análisis de varianza de la variable de respuesta tiempo de respuesta y se determina la significación de cada factor y de sus interacciones. En la cual se demuestra que los factores cantidad de servidores y mejora de capacidad, así como sus interacciones son significativos ya que sus valores  $p$  son menores que el  $\alpha$  de 0,05 y sus valores de  $F_{Tabla}$  son menores que los valores de  $F$  calculados.

Factores/Interacciones	GL	SC	MC	$F$	$P$	$F_{Tabla}$	Significación
Número Servidores (CS)	2	16,1	8,05	36,02	0	4,25	Significativo
Mejora Capacidad (MC)	2	94,30	47,15	210,98	0	4,25	Significativo
CS * MC	4	21,77	5,44	24,36	0	3,63	Significativo
Error	9	2,01	0,22				
Total	17	134,19					

Tabla 5.20: ANOVA de los factores e interacciones para la variable de respuesta Tiempo de Respuesta. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados.

Los tiempos de respuesta se presentan en la Figura 5.24, de la cual se observa: (i) con el modelo original se logra un valor *Satisfactorio* para un servidor y *Sobresaliente* para dos y tres servidores, (ii) con mejoras de 50 % y 75 % se alcanzan valores *Sobresalientes* para uno, dos y tres servidores y (iii) los menores tiempo de respuesta se dan para el caso de tres servidores con una mejora de capacidad de 75 %.

El gráfico de efectos principales (Figura 5.25) muestra sensibilidad del tiempos de respuesta frente a los cambios de nivel. Siendo mayor la disminución de dicho tiempo para el factor mejora de capacidad.

En el gráfico de interacciones para tiempos de respuesta (Figura 5.26) muestra que a medida que se incrementa el número de servidores o se mejora la capacidad se produce una disminución del mismo. Siendo significativamente mayor para el modelo original.

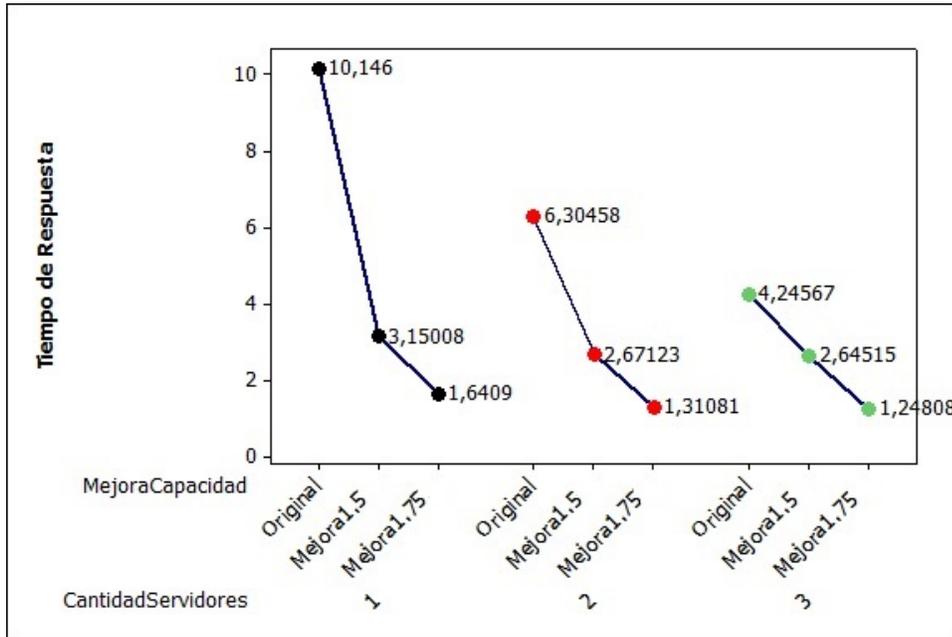


Figura 5.24: Gráfico de Tiempos de Respuesta

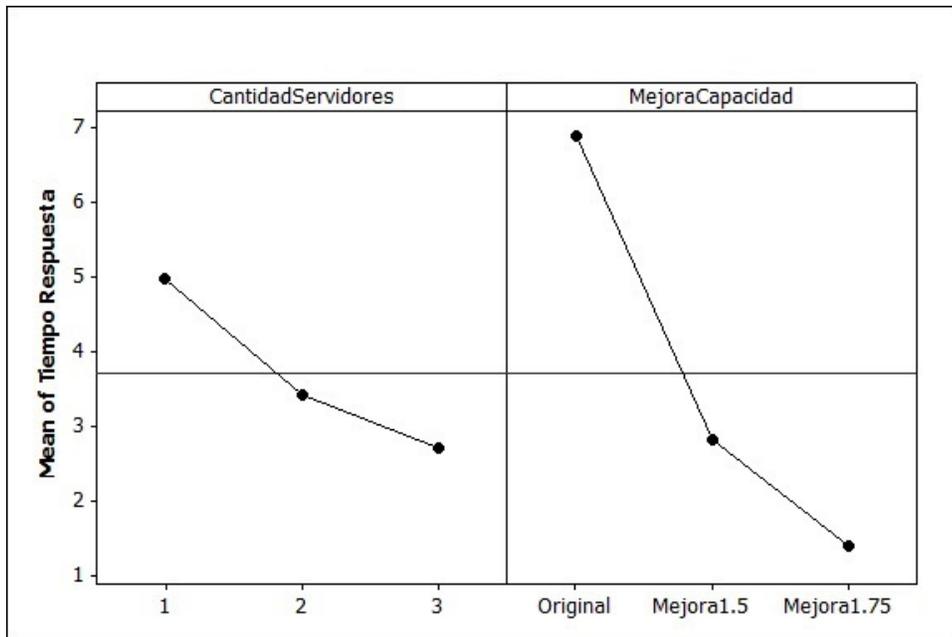


Figura 5.25: Gráfico de Efectos Principales del Tiempo de Respuesta

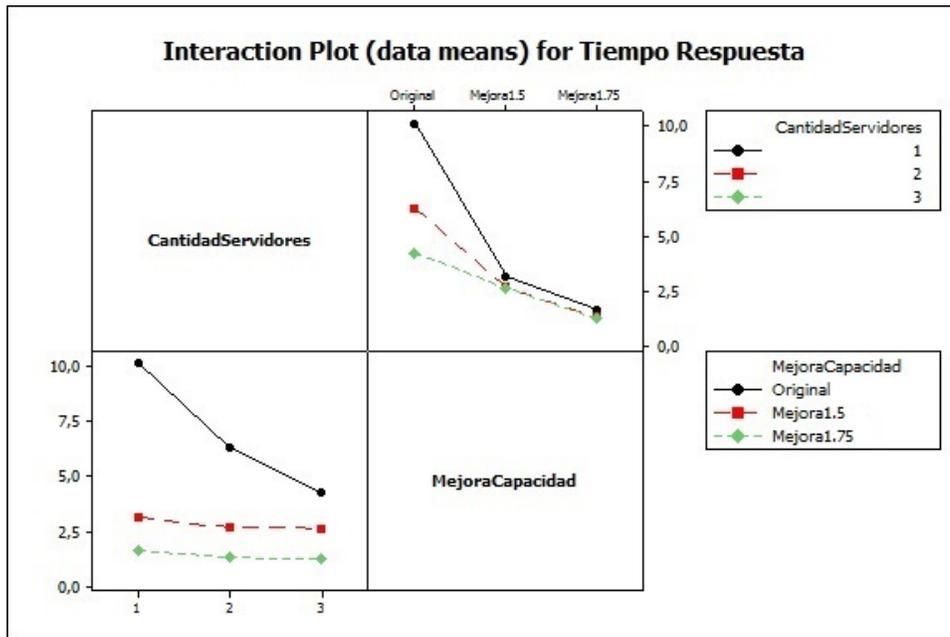


Figura 5.26: Gráfico de Interacciones del Tiempo de Respuesta

### 5.15.2. Análisis de Salidas de la Variable de Respuesta Velocidad de Procesamiento

En la Figura 5.27 se presenta el gráfico de comprobación de idoneidad del modelo para velocidades de procesamiento, del cual se concluye que se cumple la probabilidad normal de los residuos y la aleatoriedad de los datos muestreados.

En la Tabla 5.21 muestra el análisis de varianza para la variable de respuesta velocidad de procesamiento y se determina el nivel de significación de los factores y sus interacciones.

Del análisis se observa los factores cantidad de servidores y mejora de capacidad, así como sus interacciones son significativos, ya que el valor  $P$  es menor que el  $\alpha$  de 0,05 y que el  $F$  calculado es mayor que el  $F_{Tabla}$ .

En la Figura 5.28 se grafican las velocidades de procesamiento y se observa que las mayores se producen para el caso de tres servidores.

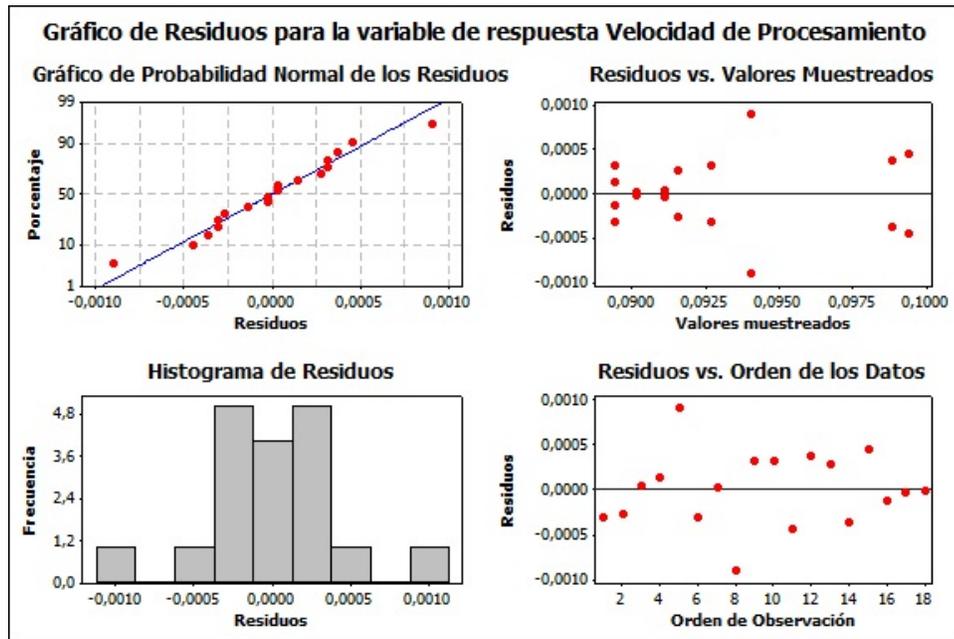


Figura 5.27: Comprobación de Idoneidad de la variable de respuesta Velocidad de Procesamiento

Factores/Interacciones	GL	SC	MC	<i>F</i>	<i>P</i>	<i>F</i> <sub>Tabla</sub>	Significación
Número Servidores (CS)	2	0,0000518	0,0000259	80,53	0	4,25	Significativo
Mejora Capacidad (MC)	2	0,0001579	0,0000790	245,52	0	4,25	Significativo
CS * MC	4	0,0000242	0,0000060	18,78	0	3,63	Significativo
Error	9	0,0000029	0,0000003				
Total	17	0,0002368					

Tabla 5.21: ANOVA de los factores e interacciones para la variable de respuesta Velocidad de Procesamiento. Siendo GL: Grados de Libertad, SC: Suma de Cuadrados y MC: Media de Cuadrados.

En el gráfico de efectos principales de la velocidad de procesamiento (Figura 5.29) se observa sensibilidad a los cambios de nivel de los dos factores, siendo mayor para el caso de mejora de capacidad.

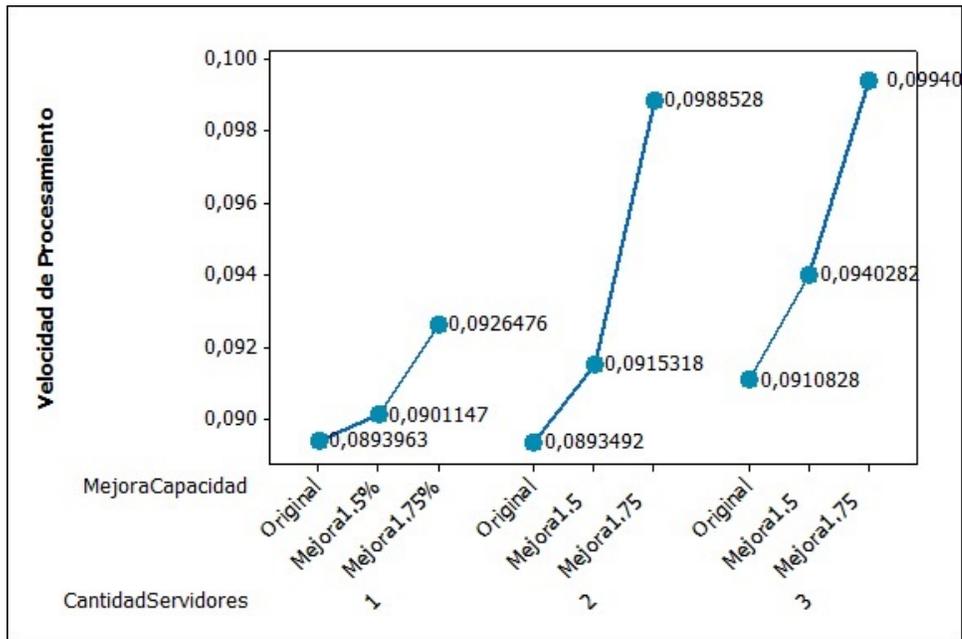


Figura 5.28: Gráfico de Velocidad de Procesamiento

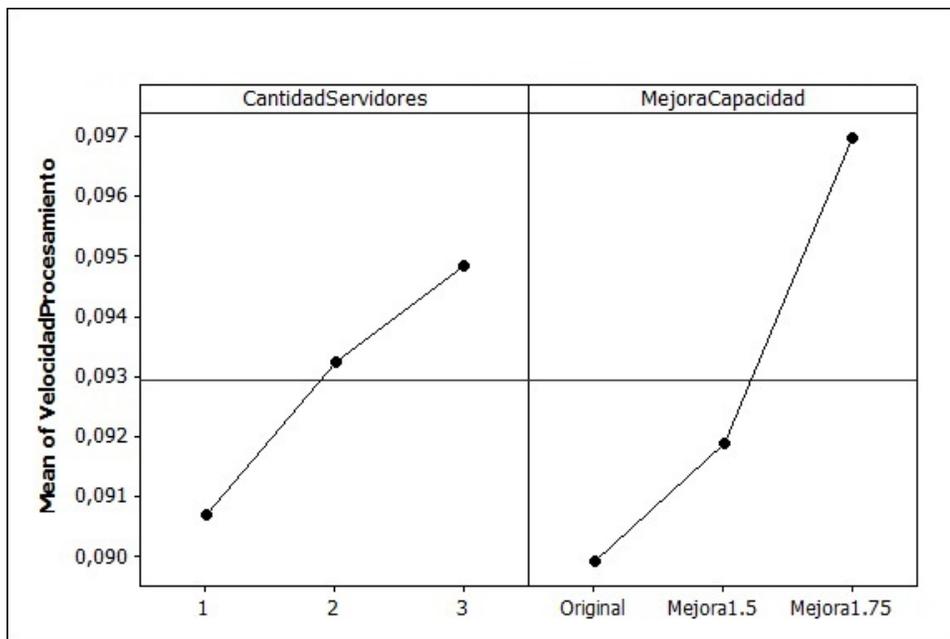


Figura 5.29: Gráfico de Efectos Principales de la Velocidad de Procesamiento

### 5.15.3. Análisis de Salidas de la Variable de Respuesta Utilización

Se presentan en la Figura 5.30 un gráfico con las utilizaciones del servidor.

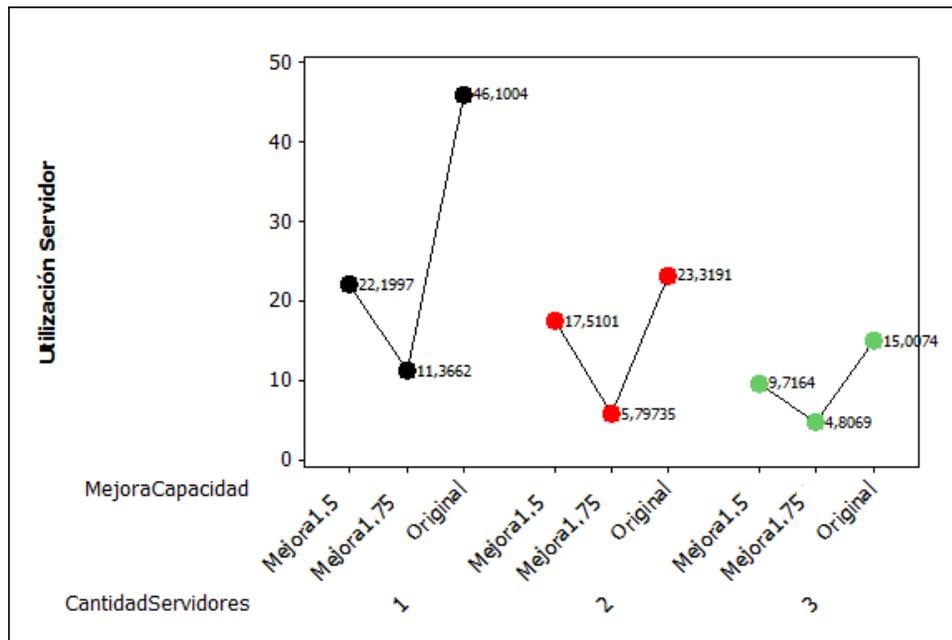


Figura 5.30: Gráfico de Utilizaciones del Servidor

La utilización del servicio Web proveedor y del servicio Web de cobro electrónico es del 0.29 %. Por tanto los recursos no muestran saturación, siendo mayores las utilizaciones para el caso del modelo original con un servidor.

Del análisis de los tiempos de respuesta, las velocidades de procesamiento y las utilizaciones se concluye que es conveniente mejorar la capacidad del servidor y elaborar un cluster con tres servidores. Si desde la perspectiva económica es factible se sugieren tres servidores como capacidad ociosa frente a picos de arribos de requerimientos no previstos.

### 5.15.4. Análisis Económico

En base a la alternativa de un cluster con tres servidores de capacidad que mejora en 75 % a la original se recalculan los costos de inversión en la Tabla 5.22, resultando un un monto total de U\$D 128.250.

Concepto	Monto (U\$D)
3 Windows Server 2012	13.650
Cisco ASA 5510 Adaptive Security Appliance	2.100
Costos del desarrollo del sitio de negocios electrónicos	112.500
<b>Costo Total de Inversión</b>	<b>128.250</b>

Tabla 5.22: Costos de Inversión para tres servidores con 75 % de mayor capacidad

Si se considera un tiempo entre arribos de 21,89 seg. se necesitan 7,36 meses para el retorno de la inversión. De acuerdo a la decisión de inversión se confirmará esta alternativa o se buscarán nuevas de menor costo.

En la Tabla 5.23 se presentan las desviaciones de los datos del diseño experimental estadístico.

Cantidad Servidores	Mejora Capacidad	Desv. Est. V. P.	Desv. Est. T.R.
1	50 %	0.00051	0.063
1	75 %	0,00044	0,098
1	Original	0,00044	0,071
2	50 %	0.0005	0.081
2	75 %	0.0005	0.014
2	Original	0.00019	0.067
3	50 %	0.0002	0.024
3	75 %	0.0006	0.076
3	Original	0.0004	0.081

Tabla 5.23: Desviación Estándar del Diseño Experimental Estadístico. V.P. Velocidad de Procesamiento. T.R. Tiempo de Respuesta.

## 5.16. Modelo de Comercio Electrónico en Capas de Servidores

En la Sección 2.3 se describió una arquitectura tradicional en capas del modelo de recursos informáticos de sistemas de comercio electrónico. En función de esta arquitectura se elabora un nuevo modelo con una plataforma informática con servidores en capas.

En la Figura 5.31 se presenta el diagrama con la definición de las principales entidades del modelo DEVS. El modelo de simulación se componen por el marco experimental, el modelo transaccional y el modelo de plataforma tecnológica.

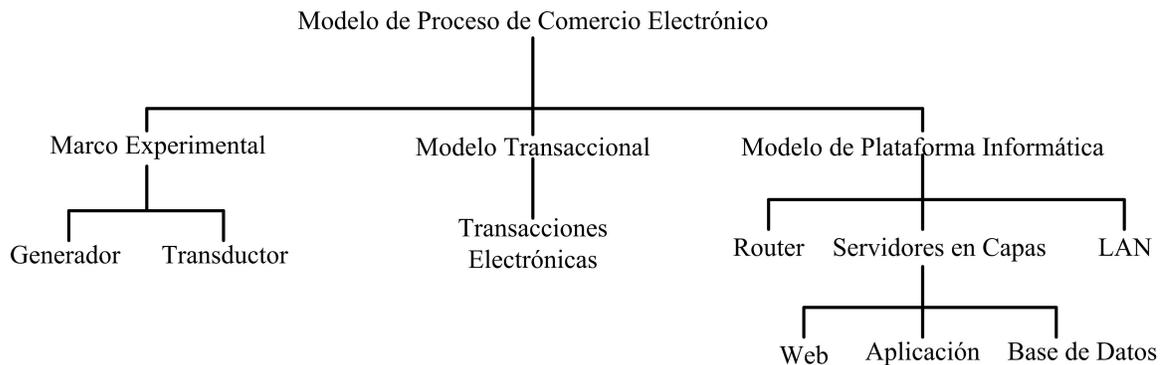


Figura 5.31: Entidades del modelo de Comercio Electrónico

El Marco Experimental y el Modelo Transaccional han sido descriptos. El modelo de Plataforma Informática se forma por un router, un grupo de servidores en capas y redes LAN (Local Area Network) que comunican los servidores. Las capas consisten en una primera de servidores Web, la segunda de servidores de aplicación y la tercera de servidores de base de datos.

En la Figura 5.32 se grafica la plataforma informática en capas, de modo que los requerimientos ingresan desde Internet por el router, se transmiten al servidor Web a

través de la LAN1, pueden pasar al servidor de aplicación por la LAN 2 y finalmente al servidor de base de datos por la LAN3.

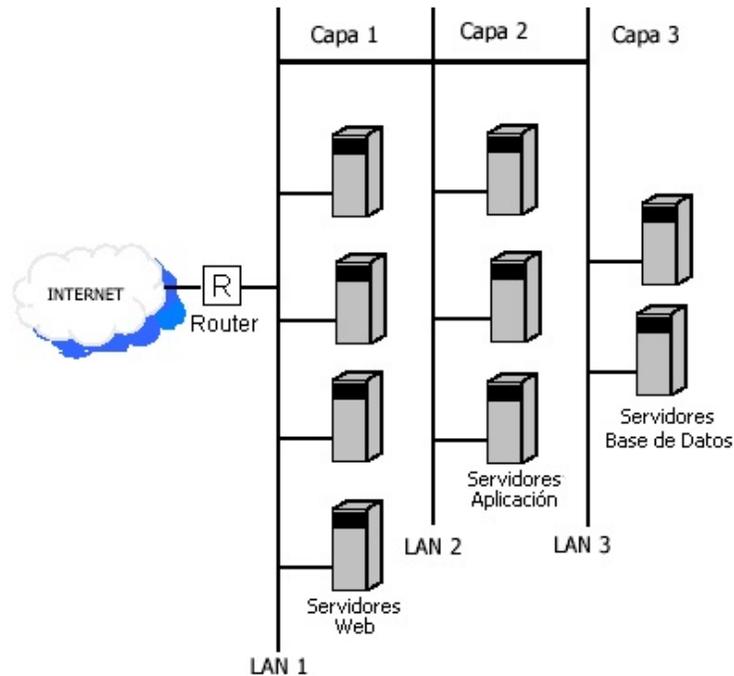


Figura 5.32: Esquema de Servidores en Capas

La arquitectura de servidores en tres capas consta de un cluster con cuatro servidores Web, tres servidores de aplicación y dos de base de datos.

Cada transacción electrónica va a tener una demanda diferente de recursos. En la Figura 5.33 se detallan los accesos a los recursos para cada transacción en un diagrama de interacción.

De acuerdo a los accesos que requieren los recursos de las diferentes capas, se definen los siguientes patrones:

- **Visita homepage:** un acceso a capa de presentación.
- **Add cart:** dos accesos a capa de presentación y un acceso a capa de aplicación.

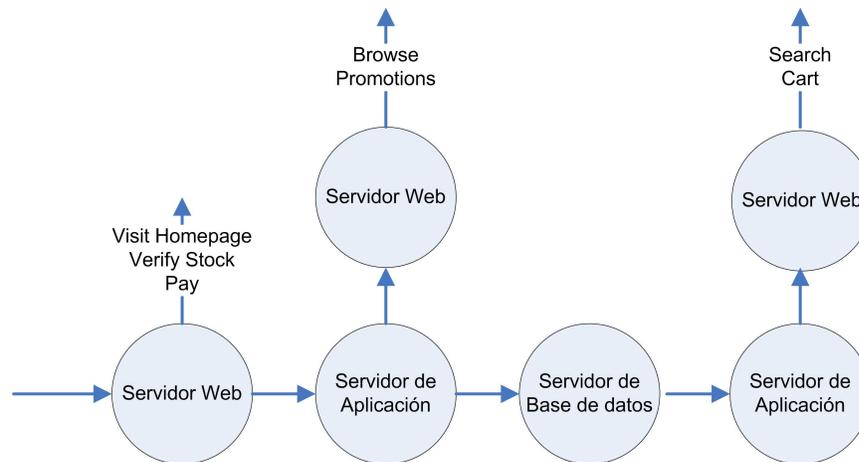


Figura 5.33: Diagrama de Interacción

- **Catalog:** dos accesos a capa de presentación, dos accesos a capa de aplicación y un acceso a capa de base de datos.
- **Select Product:** dos accesos a capa de presentación, dos accesos a capa de aplicación y un acceso a capa de base de datos.
- **Search:** dos accesos a capa de presentación, dos accesos a capa de aplicación y un acceso a capa de base de datos.
- **Menu:** dos accesos a capa de presentación y un acceso a capa de aplicación.
- **Promotions:** dos accesos a capa de presentación y un acceso a capa de aplicación.
- **Send request - Pay,** que son implementadas por los socios de negocios Proveedor y Servicio de Pagos. El acceso a éstas se realiza a través de interacciones entre los recursos informáticos de cada dominio de negocios, que se hallan integrados en una arquitectura orientada a servicios.

### 5.16.1. Modelo DEVSJAVA de la Plataforma Informática con Servidores en Capas

En la Figura 5.34 se presenta el modelo de comercio electrónico con tecnología de servidores en capas. El mismo está compuesto por los modelos DEVS acoplados Experimental Framework, Transaction y Technological Model; así como los modelos atómicos Provider y Web Service Pay. El comportamiento de los modelos acoplados Experimental Framework y Transactions no presentan variaciones a los definidos.

El modelo de la plataforma informática consiste en una arquitectura de servidores en capas. En la primera capa, se define un cluster de cuatro servidores Web, en la segunda un cluster de tres servidores de aplicación y en la tercera uno de dos servidores de bases de datos.

Los trabajos que arriban al modelo acoplado Technological Model, contienen información sobre el tipo de transacción que los envía. En base a esta información y siguiendo los patrones de utilización de recursos definidos en el diagrama de interacción, los trabajos procesados en el cluster de servidores Web pueden ser enviados a la capa de servidores de aplicación o regresar al modelo Transactions. De manera análoga, los trabajos que arriban al cluster de servidores de aplicación, luego de su procesamiento, pueden ser enviados a la capa de servidores de bases de datos o al modelo Transactions. Los trabajos procesados por los servidores de la capa de bases de datos, son enviados al modelo Transactions.

El modelo atómico Provider se corresponde con el acceso al servicio del proveedor para consultar el stock de un producto y el Web Service Pay con el servicio de cobro electrónico. Respecto al cobro electrónico se considera la tercerización del mismo, el cual autoriza y ejecuta el cobro en línea a través de una tarjeta de crédito. Para los dos tipos de servicios se consideran tecnologías de servicios Web.

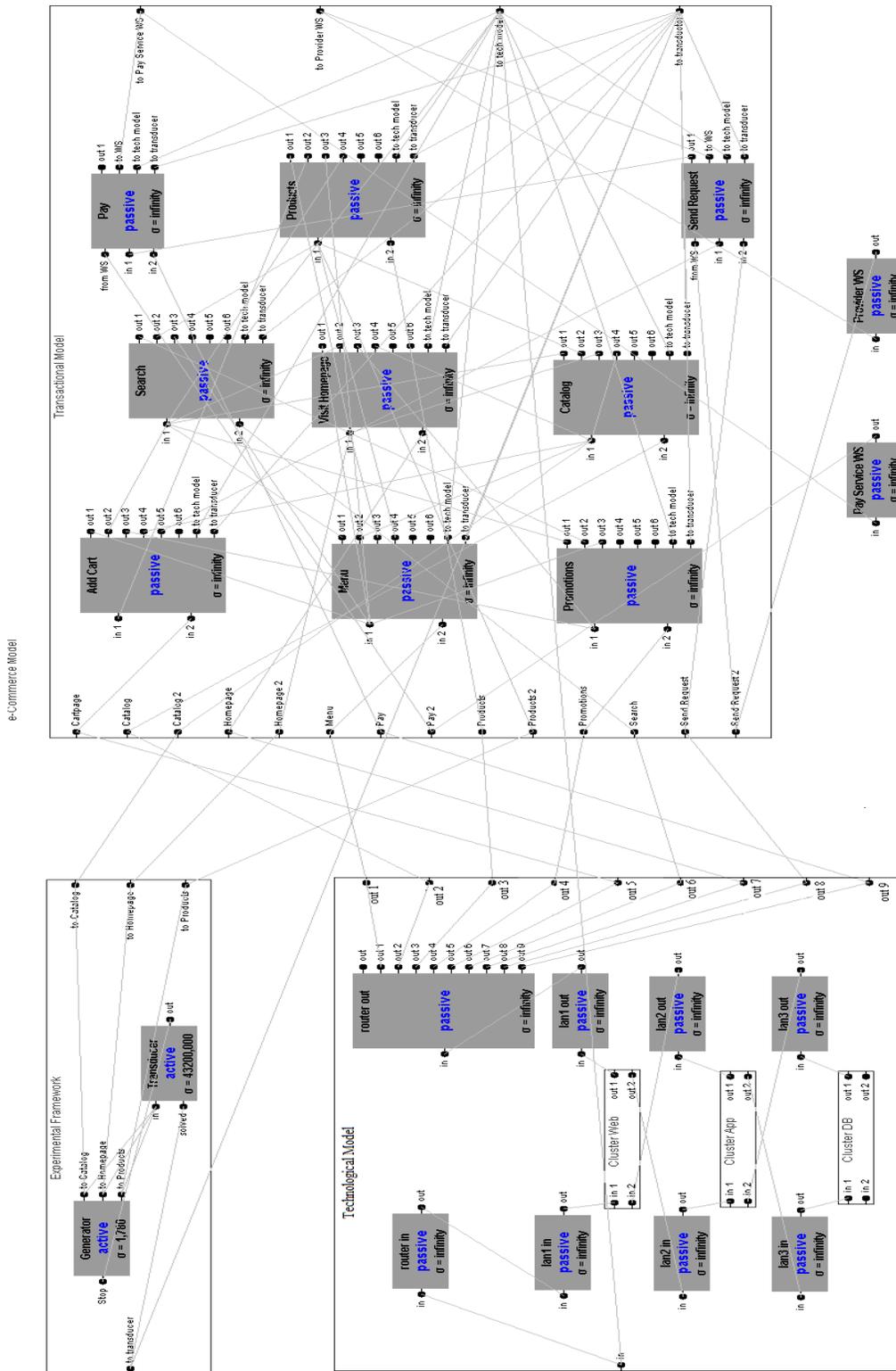


Figura 5.34: Vista del simulador DEVJSJAVA del modelo de comercio electrónico con servidores en capas

Los parámetros de tiempos de servicio de los requerimientos en cada una de las capas fueron obtenidos de mediciones efectuadas en una experiencia de laboratorio. En una computadora con procesador Core 2 DUO, 1.8 Ghz y 2 GB de RAM, con sistema operativo Linux Ubuntu 10.0.4 y un servidor de base de datos MySQL 5.1.69, se crea una tabla de libros con 100.000 registros y se realizan consultas para recuperar los datos de un libro. Del mismo modo se elaboran scripts en PHP para las transacciones restantes y se miden los tiempos de servicio promedio que se presentan en la Tabla 5.24.

Tipo de Servidor	Tiempo de servicio promedio (seg.)
Servidor de Bases de Datos	0.03
Servidor de Aplicaciones	0.75
Servidor Web	0.75

Tabla 5.24: Tiempos de servicio por servidor

Para obtener los parámetros de los servicios web de consulta de stock al proveedor y pagar se utilizó la herramienta SoapUI, de modo que para el caso del proveedor se mide el tiempo demandado en enviar un mensaje SOAP, hacer una consulta a una base de datos y recibir la respuesta. Respecto al servicio Web de pago electrónico se toma el tiempo de enviar un mensaje SOAP con los datos del cliente, autentificarlos, hacer el pago directo y recibir la confirmación de la operación. Se obtuvieron tiempos promedios de 1.06 seg.

### 5.16.2. Simulación del Modelo en Capas de Servidores

Con los parámetros anteriores, se realizaron simulaciones, con un tiempo de simulación de 24 hs, considerando los siguientes tiempos medios entre arribos: 4.38 seg., 5.47 seg., 7.3 seg., 10.95 seg. y 21.89 seg.

### 5.16.2.1. Evaluación de las Métricas de la Plataforma Informática

Como métricas de desempeño tecnológicas se toman la velocidad de procesamiento, el tiempo medio de respuesta por sesión, y la utilización de cada servidor. Los resultados se muestran en la Tabla 5.25.

Métrica	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
Tiempo entre arribo (req./seg.)	21,89	10,95	7,3	5,47	4,38
Velocidad de Procesamiento (req./seg.)	0,04718	0,091032	0,13464	0,1851	0,2287
Tiempo de Respuesta (seg.)	4,9817	5,1473	4,9664	4,9455	4,944

Tabla 5.25: Métricas tecnológicas

Se observa que a medida que se aumenta la tasa de arribos de nuevas sesiones (menor tiempo medio entre arribos), aumenta en forma análoga la velocidad de procesamiento, mientras que los tiempos medios de respuesta (totalizados por sesión de usuario), se mantienen en un valor aproximadamente constante, cercano a los 4.9 segundos. Estos hechos sugieren que la capacidad del sistema es superior a la demanda planteada.

De la Tabla 5.26, se observa que la utilización de los diferentes servidores aumenta en forma análoga a la tasa de arribos y en todos los casos es inferior al 16 %, lo cual refuerza la conclusión de que el sistema posee una capacidad superior a la necesaria. Se destaca que los servidores pertenecientes a la misma capa fueron utilizados en forma equitativa.

### 5.16.2.2. Evaluación de las Métricas de Negocios

Se busca el punto de equilibrio entre el costo de la inversión y los beneficios netos.

En la Tabla 5.27 se muestra un costo total de inversión de USD 126.790, calculado como la suma de los costos de la plataforma informática y del desarrollo del sistema de comercio electrónico.

Tiempo entre arribos (seg.)	21,89	10,95	7,3	5,47	4,38
Serv. Web 1	3,296	6,5714	9,3702	12,837	15,846
Serv. Web 2	3,296	6,5714	9,3702	12,837	15,846
Serv. Web 3	3,296	6,5713	9,3702	12,837	15,846
Serv. Web 4	3,296	6,5705	9,3702	12,836	15,846
Serv. de Aplicación 1	3,041	6,0524	8,6414	11,835	14,596
Serv. de Aplicación 2	3,041	6,0523	8,6405	11,835	14,595
Serv. de Aplicación 3	3,04	6,0515	8,6413	11,834	14,595
Serv. Base Datos	1 0,138	0,2719	0,3908	0,5359	0,6605
Serv. Base Datos 2	0,138	0,2719	0,3908	0,5359	0,6605

Tabla 5.26: Porcentaje de utilización por recurso

Costo de plataforma informática	U\$D 14,290
Cotos del desarrollo del sitio de negocios electrónicos	U\$D 112,500
Costo de Inversión	U\$D 126,790

Tabla 5.27: Costos de Inversión

En la Tabla 5.28 se detallan los datos sobre tiempos entre arribos de clientes, cantidad de clientes, productos vendidos, beneficios netos por mes. Para dichos datos se muestran los meses necesarios para cubrir los costos de inversión.

Tiempo entre arribos	Cant. clientes	Prod. vendidos	Ganancia Neta (U\$D)	Meses retorno
4,38	17806	29	29,906.25	4
5,47	15368	21	21,656.25	6
7,30	11036	17	17,531.25	8
10,95	7344	8	8,250.05	15
21,89	3849	5	5,156.25	24

Tabla 5.28: Métricas de negocios

Para disminuir el número de meses se requiere mayor número de clientes. De acuerdo al arribo de clientes actuales en el sitio se requieren dos años para cubrir la inversión. A medida que se incrementa este número se disminuyen los meses necesarios.

Por tanto se puede concluir que es tarea de marketing de la empresa la búsqueda de un incremento en el número de clientes o disminuir la cantidad de recursos informáticos

en la plataforma tecnológica de modo de bajar costos.

## 5.17. Conclusiones

Con el fin de demostrar la factibilidad de la metodología DEVS de modelización para simulación de transacciones de comercio electrónico se la aplica en un caso real de estudio de una empresa de venta minorista de electrodomésticos, electrónica, computación, artículos para el hogar, accesorios, música y películas.

Por cuestiones de privacidad no se identifica la empresa, la cual se puede considerar como grande, ya que tiene un gran número de sucursales distribuidas en diferentes ciudades de todas las provincias e importantes magnitudes de ventas.

El trabajo comienza con la exploración del sitio de comercio electrónico. De la misma se identifican las principales transacciones y la red de transiciones entre ellas.

La tarea principal consiste en construir un CBMG de las sesiones de clientes, para ello se estudia en detalle la monitorización de sesiones de clientes existente en la herramienta Google Analytics. La red de transacciones es considerablemente grande y compleja de comprender, por lo cual demanda un esfuerzo de estudio para seleccionar transacciones e interacciones que representen el real comportamiento de clientes.

El objetivo propuesto es seguir la lógica de ingreso de dinero, se seleccionan aquellas transacciones de sesiones de clientes que compran y se elabora el CBMG.

En relación a la plataforma informática, la empresa cuenta con un único servidor que implementa el Sitio de comercio electrónico. Sobre la base del modelo de cluster programado para el Capítulo 6, se lo reutiliza definiendo un único servidor. Esto es posible ya que el balanceador de carga tiene un tiempo cero de servicio. Como escenarios de mejora se incorporan nuevos servidores y se los organiza en un cluster.

Siguiendo las técnicas propuestas por la metodología se construye el modelo DEVS.

Ejecutando la simulación se contrastan los datos reales con los simulados y de este modo se valida el modelo demostrando la aplicabilidad de la metodología desarrollada.

Como resultado de la simulación se concluye que desde la perspectiva de negocios se requiere un replanteo de la estrategia comercial con la incorporación de un servicio de cobro electrónico para ventas en línea. Con esta decisión se busca el incremento de las ventas.

Respecto a la tecnología se decide evaluar nuevas plataformas y para ello se modela con servidores en capas, con una mayor especificación de los recursos informáticos.

Se concluye que es conveniente incorporar el cobro electrónico desde la perspectiva de negocios. En cuanto a la plataforma informática, con tecnología en cluster se cuenta con una solución de menor costo, pero con una de servidores en capas se dispone de capacidad ociosa para responder a ráfagas de arribos de clientes.

Los resultados de simulación muestran las bondades para simular en diversos escenarios procurando una mejora desde una perspectiva integrada de resultados tecnológicos y beneficios económicos, siendo útil no solo para la definición de configuraciones informáticas sino además una herramienta de predicción para la toma de decisión en la gestión de operaciones de comercio electrónico.

# CAPÍTULO 6

## Conclusiones y Trabajos Futuros

*En el capítulo final se realizan las conclusiones del trabajo realizado. Para ello se analiza la metodología de modelización y simulación propuesta y se resumen las principales contribuciones. Por último se proponen trabajos futuros y se presentan nuevas líneas de investigación como profundización de lo desarrollado en la tesis.*

### 6.1. Conclusiones

#### 6.1.1. Análisis de la metodología propuesta

El desafío en este trabajo consistió en desarrollar un método para diseñar procesos de comercio electrónico considerando la perspectiva tecnológica y de negocios, de modo que a partir de una estrategia comercial se estime la plataforma informática que la soporte con adecuados tiempos de respuesta y satisfactoria rentabilidad del negocio.

Como punto de partida se tomó el Modelo de Referencia de Negocios Electrónicos propuesto por Menascé y Almeida (2000), descrito en la Figura 1.4. Este modelo consiste de dos bloques, el superior correspondiente a los negocios, con metas, estrategias

y transacciones comerciales y el inferior correspondiente a la tecnología con la configuración de los recursos informáticos y la carga de trabajo según el comportamiento de los usuarios.

Con este enfoque de modelo integrando los beneficios económicos y los recursos tecnológicos se definió un modelo de negocios para implementar la metodología propuesta. Por ello en el Capítulo 2 se lo definió como una herramienta conceptual para la descripción de la estructura organizacional, establecer estrategia e identificar la lógica de ingreso de dinero.

A partir de esta definición se identificó la estrategia de negocios con el uso de tecnología informática e Internet, sus redes de socios y las transacciones. Con estos elementos se pudo construir el proceso de comercio electrónico con las secuencia de transacciones de clientes en el sitio.

Se elaboró un framework de modelización y simulación orientado a servicios detallado en la Figura 2.18, asumido como el fundamento de la metodología desarrollada y el cual determinó la arquitectura para la construcción del modelo y una referencia para la parametrización de la simulación. En base al mismo, se elaboró la metodología de modelización y simulación orientada a servicios de procesos de comercio electrónico que se puntualiza en la Figura 6.1 y de la cual se resume:

- A nivel de negocios se formula su estrategia y se plantea el modelo a realizarse con los recursos tecnológicos, tales como cartelera publicitaria, catálogo, venta en línea con cobro telefónico, venta en línea con cobro totalmente virtual y servicios al cliente entre otras.
- Definidos la estrategia y el modelo de negocios se plantea el proceso de negocios. Para ello se proponen las transacciones que implementan la estrategia comercial, se elabora la coreografía de interacciones con socios y la orquestación de transac-

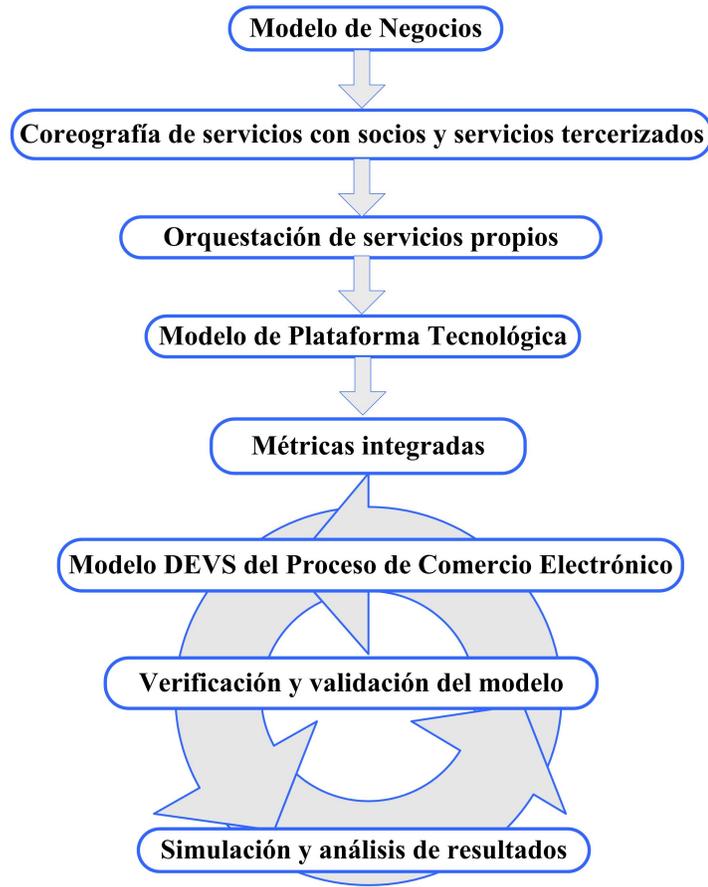


Figura 6.1: Etapas de la metodología de modelización para simulación

ciones internas. Para la modelación de la coreografía se utiliza el diagrama de secuencia UML y para la orquestación se elabora el modelo de comportamiento de los clientes mediante el gráfico CBMG. Un modelo CBMG representa las sesiones para una clase de clientes navegando en el sitio. Este gráfico indica las transacciones del sitio de negocios electrónicos como los estados posibles del sistema y los caminos a seguir por el cliente como las transiciones entre estados de acuerdo a una probabilidad. Para su construcción se utilizan herramientas de monitorización de sitios reales de comercio electrónico. A partir de esta monitorización se obtiene, de la cantidad total de clientes por transacción, la proporción que toma cada puerto de salida y de este modo se calcula la probabilidad de seguir

a la próxima transacción.

- Una vez diseñado el proceso de comercio electrónico se propone la plataforma tecnológica informática con la identificación de los recursos computacionales. Por último se plantean las métricas integradas de desempeño del modelo a nivel de tecnología y negocios.
- Para elaborar el modelo de comercio electrónico se debe identificar cada componente del sistema y sus relaciones. En DEVS cada componente se identifica con un modelo atómico y sus relaciones con los acoplamientos. Por ello, a través del uso de un diagrama de estructura se propone un nodo raíz correspondiente al modelo de comercio electrónico del cual se desprenden dos ramas, el marco experimental y el sistema de comercio electrónico. El marco experimental contiene el generador de cargas de trabajo y el transductor que computa las métricas de salida. El sistema de comercio electrónico se compone del modelo transaccional que representa las operaciones electrónicas de negocios y la plataforma tecnológica que comprende los recursos informáticos.
- Por cada modelo atómico se construye un diagrama statechart que identifica sus puertos de entrada y salida, el conjunto de estados y las funciones de transición de estados. De este modo se elabora la semántica del formalismo DEVS.
- Se obtienen los parámetros de las variables de entrada y de los recursos tecnológicos. Las cargas de trabajos se parametrizan mediante distribuciones de probabilidad que representan el comportamiento de arribos de clientes y para los recursos se estima el tiempo medio de servicio de cada transacción procesada en los mismos. Esta información se adquiere a través de monitorizaciones y tratamiento de los datos recolectados de sitios reales o emulados.

- Se implementa el modelo en la herramienta de simulación DEVSJAVA, se verifica su lógica y se lo valida con datos reales.
- Con el modelo validado se proponen los diferentes escenarios de experimentación para ejecutar la simulación. En las últimas tres etapas se pueden producir ciclos retornando a las anteriores.

La metodología propone técnicas para la construcción del modelo de comercio electrónico con base en una estrategia de negocios, la obtención de sus parámetros y correspondiente simulación.

Su principal ventaja es la capacidad para integrar métricas de tipo tecnológicas con las de negocios, de modo que a partir de la configuración de una plataforma informática se puedan obtener medidas del tiempo de respuesta por un lado e ingresos por ventas y retorno a los costos de inversión por otro.

A través del diagrama de estructuras y el gráfico statechart se dispone de una técnica para la construcción de los algoritmos y de la especificación del formalismo DEVS que define el modelo.

Con las capacidades del formalismo DEVS de modelado jerárquico, modular y con orientación a objetos, se construyen modelos de simulación basados en jerarquía de componentes, escalabilidad y reusabilidad.

La simulación del modelo se realiza en DEVSJAVA, que consiste en una herramienta de software orientada a objetos, de libre disponibilidad para aplicaciones académicas con acuerdo de licencias. El lenguaje de programación es JAVA y se implementan modelos expresados según el formalismo DEVS.

La selección de esta herramienta se realiza considerando también que el método de simulación es independiente del modelo, de modo que se construye un modelo de sistema en cuestión sin estar sujeto a las restricciones de un simulador.

Respecto a la metodología propuesta una desventaja importante es el tiempo de experimentación para la simulación, con una carga horaria considerable para las corridas.

Por otro lado, el procesamiento de los datos obtenidos de las salidas de simulación que se registran en los archivos de Logs, se realiza mediante otras aplicaciones informáticas, lo cual implica un proceso no automatizado de tratamiento de resultados.

Otro problema es la parametrización de los recursos informáticos, ya que el tiempo de servicio obtenido depende de la aplicación particular, del sistema operativo y de la computadora sobre la cual se ejecuta. Por tanto los resultados obtenidos son aplicables estrictamente al escenario particular de simulación y solo como una aproximación para otros recursos similares.

Respecto a los casos de proceso de comercio electrónico, se trabaja en base a estrategias de venta minorista de productos. Primero se construye un modelo para la venta de libros en línea con implementación de operaciones totalmente virtual y luego un caso de estudio real para la venta en línea de electrodomésticos, electrónica, computación, artículos para el hogar, accesorios, música y películas, con ejecución de la compra en forma telefónica y su extensión a tarjeta de crédito.

Por tanto a las transacciones propias del sitio de negocios se anexan dos relacionadas con el cobro electrónico y la consulta de inventario al proveedor. De este modo se propone una coreografía de integración de servicios con socios y una orquestación de transacciones internas, logrando así modelos de aplicaciones B2C con integración a B2B.

Para ejecutar el modelo de proceso de comercio electrónico se representan modelos de plataformas informáticas en base a un cluster de servidores en paralelo y a la arquitectura tradicional en capas de servidores Web, aplicación y base de datos. Al cluster se lo modela como un grupo de servidores coordinados por un balanceador de cargas que distribuye los requerimientos en forma secuencial en los servidores desocupados y se considera la existencia de fallas en los mismos con la posibilidad de recuperación.

### 6.1.2. Principales Contribuciones

Las principales contribuciones se describen en función de los trabajos presentados en congresos, conferencias y jornadas de ciencia y tecnología.

Del proceso de investigación llevado adelante se ha desarrollado una metodología de modelado y simulación de procesos de comercio electrónico sobre la base del formalismo DEVS, con aplicaciones específicas en sistemas estocásticos y con paralelismo, la cual se describe en el trabajo de Chezzi y otros (2013). A partir de dicha metodología se construyen modelos de proceso de comercio electrónico para su simulación.

La característica fundamental es la evaluación integrada de métricas, de modo que con base en una plataforma informática se mide su desempeño, se calculan los costos de inversión y se los compara con los beneficios económicos a obtener. La metodología se prueba en un caso de estudio real para un empresa minorista de electrodomésticos y artículos del hogar de alcance nacional. Para el caso de estudio se diseña un modelo CBMG de comportamiento de clientes diferente a los existentes en la bibliografía.

En la construcción de la metodología se ha detectado la necesidad de determinar métricas de evaluación de desempeño a nivel de negocios y tecnología con el fin de realizar un análisis integrado que responda a: “¿qué recursos informáticos se necesitan para llevar adelante el proceso de comercio electrónico?”, “¿cuál es su tiempo de respuesta y disponibilidad?” y “¿en cuánto tiempo se obtiene un retorno de la inversión?”. Por ello en el trabajo de Chezzi y otros (2008) se propone un framework de métricas que integran los requerimientos de negocios y tecnológicos.

Para encontrar las respuestas a las preguntas formuladas con anterioridad se propone un algoritmo de búsqueda de configuraciones de sistemas eficientes de comercio electrónico. De este modo, para una configuración de negocios y de arquitectura informática se obtiene una solución adecuada medida por el punto de equilibrio entre los

beneficios y los costos de inversión en una plataforma informática que alcanza tiempos de respuesta óptimos.

De la exploración de trabajos científico-tecnológicos se observa que son las arquitecturas orientadas a servicios las utilizadas para el diseño de procesos de comercio electrónico en la actualidad y con una tendencia creciente. Por ello en la metodología propuesta se incorpora la capacidad de modelar este tipo de arquitecturas.

Como técnica de análisis de las salidas de simulación se trabaja con el diseño experimental estadístico, para obtener las variables de respuesta e identificar el efecto de los factores de entrada en las mismas. Para ello se implementa un caso en el trabajo de Chezzi y otros (2007), en el cual se muestra el uso de un diseño factorial completo para la evaluación de salidas de simulación.

El uso de la herramienta de simulación DEVJSJAVA permite obtener una vista de los componentes atómicos o acoplados y su interacción a través del paso de mensajes durante la simulación, por lo cual es posible observar la secuencia de las corridas. De este modo se cuenta con una facilidad para el aprendizaje del formalismo DEVS, control de diseño y verificación del modelo.

Si bien la complejidad de programación en DEVS puede ser mayor que en un programa específico de simulación, la decisión de su uso se fundamenta en la factibilidad de construir modelos sin restricciones de funcionalidades.

Por último se amplía la aplicación de la metodología en el trabajo de Cámara y otros (2013), en el cual se cambia la configuración de tecnología informática por un modelo en capas de servidores Web, aplicación y base de datos.

En resumen se destacan las siguientes contribuciones:

- **Formulación de antecedentes conceptuales de negocios y tecnología:** recopilación y análisis bibliográfico como fundamento para esbozar los conceptos

de modelos de negocios y de procesos de negocios en relación con los beneficios económicos de la implementación de sistemas de comercio electrónico.

- **Framework de métricas integradas de negocios y tecnología:** propuesta de métricas a nivel de tecnología, negocios y de integración entre ellas. Con el mismo, se cuenta con un modelo de métricas para la evaluación de desempeño de una configuración informática y su capacidad de producir beneficios.
- **Framework de modelización y simulación orientado a servicios:** construcción de una abstracción de los componentes necesarios para la modelización y simulación de sistemas de comercio electrónico orientados a servicios.
- **Metodología de modelización y simulación de procesos de comercio electrónico:** formalización de una metodología a través de etapas sistematizadas para la construcción del modelo DEVS y su parametrización, simulación y evaluación de resultados.
- **Incorporación de componentes estocásticos en la simulación DEVS:** el modelo transaccional sigue un comportamiento aleatorio de modo que con su implementación se muestra las capacidades para simulación estocástica con el formalismo DEVS.
- **Algoritmo de obtención de configuraciones eficientes de tecnología y negocios:** desarrollo de una técnica para la búsqueda de configuraciones de plataformas informáticas en relación con las posibilidades de retorno de la inversión.
- **Modelo CBMG de comportamiento de clientes:** obtención de un modelo CBMG de clientes para el caso de estudio diferente a los existentes.

- **Modelos de procesos de comercio electrónico:** propuesta de modelos de comercio electrónico para la implementación de la metodología propuesta.

## 6.2. Trabajos Futuros

- **Considerar el tiempo total de respuesta desde que el cliente inicia un requerimiento hasta que realiza el próximo:** Al incorporar al cliente en el modelo se debe considerar su tiempo de reflexión o pensado. Los modelos hasta acá desarrollados estimaron el tiempo de respuesta para sesiones con secuencia continua de transacciones. En el caso de aplicar el tiempo de pensado o de reflexión, la plataforma informática podrá atender otros requerimientos. El tiempo de respuesta será el que percibe el cliente, ya que se considera la demora entre las decisiones para las diferentes transacciones.
- **Desarrollar herramientas de simulación a partir de la metodología propuesta para posibilitar:**
  1. Construcción de los componentes del modelo a través de un entorno gráfico, de modo que a través de la incorporación de modelos atómicos y sus acoplamientos se elaboren en forma interactiva los modelos de procesos de comercio electrónico.
  2. Entorno de configuración de escenarios de experimentación y presentación de salidas: incorporar a la herramienta la posibilidad de efectuar diseños experimentales estadísticos sobre el modelo parametrizado y mostrar resultados en forma gráfica y numérica.
  3. Repositorio de experimentaciones: contar con un sistema de base de datos

que almacene los resultados de los experimentos para la comparación de salidas entre escenarios.

- **Modelo de procesos:** desde la perspectiva de negocios se trabajó sobre modelos de proceso de comercio electrónico con transacciones únicas por cada operación de negocios. Es decir, el modelo CBMG contiene las transacciones de negocios como operaciones atómicas. Considerando conceptos de procesos de negocios se podría profundizar la construcción del modelo a través de la explotación en subprocesos y especificaciones en mayor detalle de las operaciones.
- **Especializar la transacción *agregar producto al carro de compras*:** la gestión actual del carro de compras consiste en agregar un producto por cliente. Para una aproximación mayor a experiencias de la realidad se debe reformular la transacción de modo que por cada cliente se pueda mantener una lista de productos y sus precios. Es decir, los clientes pueden seleccionar uno o más productos en su carro de compras, pueden hacerlo en una sola vez o en varias visitas al sitio manteniendo o cambiando su carro de compras, con un precio que puede ir variando. Además se puede calcular los ingresos por ventas en el proceso de simulación.
- **Incorporar nuevas estrategias comerciales:** En la Sección 2.2 se plantearon categorías de procesos de comercio electrónico de acuerdo al tipo de transacción que realizaban, sobre la base de estas categorías se pueden plantear nuevos modelos, agregando funcionalidades tales como las ofrecidas por las redes sociales, tecnologías móviles, subastas, entornos de aprendizaje virtual y gestión en organizaciones gubernamentales.
- **Implementar nuevas métricas que contemplen acciones de negocios, tales como:**

- Pérdidas económicas por mal desempeño de la plataforma informática.
  - Clientes que habiendo agregado productos al carro de compras no concretan la operación.
  - Transacciones más visitadas por los clientes.
  - Efectos de la publicidad en el incremento de clientes.
  - Nivel de confianza en las operaciones ofrecidas en el sitio.
- 
- **Extender la aplicación de la metodología a modelos de servicios cloud computing (en castellano computación en la nube):** elaborar modelos de plataformas tecnológicas en este nuevo estilo arquitectural y analizar el desempeño para evaluar su incorporación en los procesos de comercio electrónico.
  - **Considerar la evaluación del desempeño del software:** extender la evaluación al desempeño del sistema informático que ejecuta el proceso de comercio electrónico.

De los trabajos realizados y futuros se definen las siguientes líneas de investigación:

- Modelado y simulación de negocios electrónicos orientado a servicios para una evaluación integrada de métricas de negocios y tecnología.
- Diseño de modelos de comercio electrónico en arquitecturas distribuidas para la evaluación del desempeño tecnológico y la predicción de los resultados económicos.
- Modelización del proceso de ventas para la especialización del modelo de negocios.
- Especificación de modelos de procesos de comercio electrónico para diferentes estrategias tecnológicas.

## Bibliografía

- ABUOSBA, K. y EL-SHEIKH, A. (2008). «Formalizing Service-Oriented Architectures». *IT Professional*, **10(4)**, pp. 34–38.
- AGRALI, S.; TAN, B. y KARAESMEN, F. (2008). «Modeling and Analysis of an Auction-based Logistics Market». *European Journal of Operational Research*, **191(1)**, pp. 272–294.
- ALAM, F.M.; MCNAUGHT, K.R. y RINGROSE, T.J. (2004). «A Comparison of Experimental Designs of a Neuronal Network Simulation Metamodel». *Simulation Modelling Practice and Theory*, **12(7-8)**, pp. 559–578.
- ALMEIDA, V.A.F. (2002). «Capacity Planning for Web Services Techniques and Methodology». *Computer Science*, **2459**, pp. 293–302.
- ALMEIDA, V.A.F. y MENASCE, D.A. (2002). «Capacity planning an essential tool for managing Web services». *IT Professional*, **4(4)**, pp. 33–38.
- ANDREOLINI, M.; CARDELLINI, V. y COLAJANNI, M. (2002). «Benchmarking Models and Tools for Distributed Web-Server Systems». *Performance Evaluation*, pp. 208–235.
- BACIGALUPO, D.A.; JARVIS, S.A.; HE, L. y NUDD, G.R. (2005). «An Investigation

- into the Application of Different Performance Prediction Techniques to e-Commerce Applications». *The Journal of Supercomputing*, **346(2)**, pp. 93–111.
- BALBO, G.; DESEL, J.; JENSEN, K.; REISIG, W.; G., ROZENBERG. y SILVA, M. (2000). «Introductory Tutorial Petri Nets». En: *In Proc. 21st International Conference on Application and Theory of Petri Nets*, .
- BALDWIN, R.O.; DAVIS IV, N.J.; MIDKIFF, S.F. y KOBZA, J.E. (2003). «Queueing network analysis: concepts, terminology, and methods». *The Journal of Systems and Software*, **66(2)**, pp. 99–117.
- BARTÉS, X. CINTAS P., A. LLABRÉS y FERNÁNDEZ, L. (2000). *Métodos Estadísticos. Control y Mejora de la Calidad*.
- BASU, A. y MUYLLE, S. (2011). «Assessing and enhancing e-business processes». *Electronic Commerce Research and Applications*, **10(4)**, pp. 437–499.
- BREMSER, W. y CHUNG, Q. (2005). «A framework for performance measurement in the e-business environment». *Electronic Commerce Research and Applications*, **4(4)**, pp. 395–412.
- CACE (2012). «Estudio de Comercio Electrónico Argentina 2012». En: <http://www.cace.org.ar/estadisticas/>, pp. 1–18. Último acceso 10 de septiembre de 2013, 15 h.
- CAO, X. (1989). «A Comparison of the Dynamics of Continuous and Discrete Event Systems». En: *In Proc. IEEE*, volumen 77, pp. 7–13. IEEE.
- CASTRO, RODRIGO; ERNESTO, KOFMAN y WAINER, GABRIEL. (2010). «A Formal Framework for Stochastic Discrete Event System Specification Modeling and Simulation». *Simulation*, **86(10)**, pp. 587–611.

- CASTRO LEON, E. y JACKSON, H. (2009). «Virtual Service Grids: Integrating IT with Business Processes». *IT Professional*, **11(3)**, pp. 7–11.
- CHEN, A.; SEN, K.N.; SAGNIKA, S. y BENJAMIN, B. (2006). «Strategies for Effective Web Services Adoption for Dinamic e-Businesses». *Decision Support Systems*, **42(2)**, pp. 789–809.
- CHEN, M.Y.; HUANG, M.-J. y CHENG, Y.-C. (2009). «Measuring knowledge management performance using a competitive perspective: An empirical study». *Expert Systems with Applications*, **36(4)**, pp. 8449–8459.
- CHENEY, P. H. y DICKSON, G. W. (1982). «Organizational Characteristics and Information Systems: An Exploratory Investigation». *Academy of Management Journal*, **25(1)**, pp. 170–182.
- CHEZZI, C.M.; TYMOSCHUK, A.R. y LERMAN, R. (2013). «A Method for DEVS Simulation of E-Commerce Processes for Integrated Business and Technology Evaluation». En: *In Proc. 2013 Spring Simulation Multi-Conference. Symposium on Theory of Modeling and Simulation - DEVS Integrative MS Symposium (TMS/DEVS)*, pp. 404–409. ACM.
- CHEZZI, C.M.; TYMOSCHUK, A.R. y VILLAMONTE, A. (2007). «Modelo de Simulación de Transacciones Electrónicas Comerciales». En: *In Proc. I Encuentro Regional Argentino Brasileño de I.O. XXI ENDIO Encuentro Nacional de Docentes en I.O. XIX EPIO Escuela de Perfeccionamiento en I.O.*, .
- CHEZZI, C.M.; TYMOSCHUK, A.R. y VILLAMONTE, A. (2008). «Herramienta para Análisis de Configuraciones de Sistemas Eficientes de e-Business». En: *In Proc. XXIV Conferencia Latinoamericana de Informática (CLEI)*, .

- CHU, S.-C.; C., LEUNG L.; HUI, Y. V. y CHEUNG, W. (2007). «Evolution of e-commerce Web sites: A conceptual framework and a longitudinal study». *Information Management*, **44(2)**, pp. 154–164.
- CHUNG, W. y TSENG, T.-L. (2012). «Discovering business intelligence from online product reviews: A rule-induction framework». *Expert Systems with Applications*.
- CLAFFY, K.; MEINRATH, S. D. y BRADNER, S. O. (2007). «The (un)Economic Internet?» *IEEE Internet Computing*, **11(3)**, pp. 53–58.
- CÁMARA, J. M.; LERMAN, R.; BARONI, F. y CHEZZI, A. R., C. MA. AND TYMOSCHUK (2013). «Modelo de Simulación de Sistema de Comercio Electrónico para Evaluación de Desempeño». En: *In Proc. 42 Jornadas de Informática e Investigación Operativa (JAIIO 2013). Simposio de Investigación Operativa (SIO 2013)*, .
- D AMBROGIO, A.; IAZEOLLA, G. y PASINI, L. (2007). «A Method for the Production of Simulation Models with Application to Web Interaction Paradigms». *Simulation Modelling Practice and Theory*, **15(5)**, pp. 605–620.
- DATLA, V. y GOSEVA-POPSTOJANOVA, K. (2005). «Measurement-based Performance Analysis od E-commerce Applications with Web Services Component». En: *In Proc. IEEE International Conference on s-Business Engineering*, pp. 305–314. IEEE.
- DE GENTILI, EMMANUELLE; DE CICCIO, ANGE y SANTUCCI, JEAN FRANCOIS. (2005). «DEVS and Fuzzy Logic to Model and Simulate a Manufacturing Process». *Computer Society*, **1**, pp. 557–564.
- DEMIRKAN, H.; KAUFFMAN, R.; VAYGHAN, J.; FILL, H.-G.; KARAGIANNIS, D. y MAGLIO, P. (2008). «Service-oriented technology and management: Perspectives on

- research and practice for the coming decade». *Electronic Commerce Research and Applications*, **7(10)**, pp. 356–376.
- DEZHGOSHA, K. y ANGARA, S. (2005). «Web services for designing small-scale Web applications». En: *In Proc. IEEE International Conference on Electro Information Technology*, IEEE.
- DOGANOVA, L. y EYQUEM-RENAULTB, M. (2009). «What do business models do?. Innovation devices in technology entrepreneurship». *Research Policy*, **38(10)**, pp. 1559–1570.
- EBUSINESS REPORT, THE EUROPEAN (2011). «ICT and eBusiness for an Innovative and Sustainable Economy». *Informe tico*, eBusiness W@tch.
- FAOUR, A. y MANSOUR, N. (2006). «Weblins: A Scalable WWW Cluster-based Server. Advances in Engineering Software». *Advances in Engineering Software*, **37(1)**, pp. 11–19.
- FASIHUL, A.; MCNAUGHT, K. y RINGROSE, T. (2004). «A Comparison of Experimental Designs in the Development of a Neuronal Network Simulation Metamodel». *Simulation Modelling Practice and Theory*, **12(2)**, pp. 559–578.
- FENG, W. y ZHANG, Y. (2008). «A birth–death model for Web cache systems: Numerical solutions and simulation». *Nonlinear Analysis. Hybrid Systems*, **2**, pp. 272–284.
- FOLAN, P. y BROWNE, J. (2005). «A Review of Performance Measurement: Towards Performance Management». *Computer in Industry*, **56(7)**, pp. 663–680.
- FRANCS, C.; DA LUZ OLIVEIRA E. J., J., COSTA; SANTANA, M.; BRUSCHI, S. y CARVALHO, S. (2005). «Performance Evaluation Base on System Modeling using

- Statecharts Extensions». *Simulation Modelling Practice and Theory*, **13**, pp. 584–618.
- GANG, N. (2008). «A Scheme of Workflow Management System Based on Web Services». En: *In Proc. of the International Symposium on Electronic Commerce and Security*, pp. 53–56. ACM.
- GARCIA, J.; VANDERDONCKT, J.; LEMAIGE, C. y CALLEROS, J. (2008). «How to Describe Workflow Information Systems to Support Business Process». En: *In Proc. of the 10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services*, pp. 404–411. IEEE.
- GIAGLIS, G.; RAY, P. y DOUKIDIS, G. (1999). «Assessing the Impact of Electronic Commerce on Business Performance: A Simulation Experiment». *Electronic Markets*, **9(1)**, pp. 25–31.
- GOMORY, S.; HOCH, R.; LEE, J.; PODLASECK, M. y SCHOMBERG, E. (1999). «E-Commerce Intelligence: Measuring, Analyzing and Reporting on Merchandising Effectiveness of Online Stores». *Informe tico*, IBM Research Report.
- GREASLEY, A. (2000). «Effective uses of business process simulation». En: *In Proc. of the 2000 Winter Simulation Conference*, pp. 2004–2009.
- GUIDO, C.; LUCCHI, R. y MAZZARA, M. (2007). «A Formal Framework for Web Services Coordination», **180**, pp. 55–70.
- GUO, K., J. A CHUN FOO; BARBOUR, L. y ZOU, Y. (2008). «A Business Process Explorer: Recovering and Visualizing E-Commerce Business Processes». En: *In Proc. of the 30th International Conference on Software Engineering*, pp. 871–874. ACM.

- HAHN, J.; KAUFFMAN, R. J. y PARK, J. (2002). «Designing for ROI: Toward a Value-Driven Discipline for E-Commerce Systems Design». En: *35th Hawaii International Conference on System Sciences*, .
- HO, S.-C.; KAUFFMAN, R. y LIANG, T.-P. (2007). «A growth theory perspective on B2C e-commerce growth in Europe: An exploratory study». *Electronic Commerce Research and Applications*, **6**, pp. 237–259.
- HOLLOCKS, B. (2001). «Discret-event simulation: an inquiry into user practice». *Simulation Practice and Theory*, **8(6-7)**, pp. 451–471.
- HOOK, GEOFFREY. (2011). «Business Process Modeling and Simulation». En: *In Proc. of the 2011 Winter Simulation Conference*, pp. 2004–2009.
- HORNGREN, C.T. (1985). *Contabilidad de Costos. Un enfoque de Gerencia*. Ed. Pentrice-Hall Hispanoamericana S.A..
- HUANG, C.-Y.; TZENG, G.-H. y HO, W.-R. JERRY (2011). «System on chip design service e-business value maximization through a novel MCDM framework». *Expert Systems with Applications*, **38(7)**, pp. 7947–7962.
- HUANG, J.-M.; JOU, ZHANG L.-C. WANG S.-T., Y.-T. y HUANG, C.-X. (2009). «A web-based model for developing: A model base design system». *Expert Systems with Applications*, **36(4)**, pp. 356–367.
- JAIN, R. (1991). *The Art of Computers Systems Performance Analysis*. Ed. John Wiley Sons.
- KAPLAN, J. y HUGHES, J. (2009). «Where IT infrastructure and business strategy meet». *Informe tico*, The McKinsey Quarterly.

- KHOSHNEVIS, BEHROKH (1994). *Discrete Systems Simulation*.
- KIM, J. y HUEMER, CH. (2004). «From an ebXML BPSS choreography to a BPEL-based implementation». *ACM SIGecom Exchange*, **5(2)**, pp. 1–11.
- KIM, SUNGUNG.; SARJOUGHIAN, HESSAM S. y ELAMVAZHUTHI, VIGNESH. (2009). «A Simulator Supporting Visual Experimentation Design and Behavior Monitoring.» En: *In Proc. of the 2009 Spring Simulation*, ACM.
- KLEINROCK, L. (1993). «On the Modeling and Analysis of Computer Networks.» En: *In Proc. IEEE*, volumen 81, pp. 1179–1191. IEEE.
- KULKARNI, NAVEEN; KUMAR, KUMARASAMY; MANI, SENTHIL y PADMANABHUNI, SRINIVAS. (2005). «Web Services: E-Commerce Partner Integration.» *IT Professional*, **7(2)**, pp. 23–28.
- LAI, V. (2001). «Intraorganizational communication with Intranets». *Communications of the ACM*, **44(7)**, pp. 95–100.
- LAZOWSKA, E.; ZAHORJAN, J.; GRAHAM, S. y SEVCIK, K. (1984). *Quantitative System Performance. Computer System Analysis Using Queueing Network Models*.
- LEE, Y. y KOZAR, K. (2006). «Investigating the effect of website quality on e-business success: An analytic hierarchy process (AHP) approach». *Decision Support Systems*, **42(3)**, pp. 1383–1401.
- LIN, K.-J. (2008). «E-Commerce Technology: Back to a Prominent Future». *IEEE Internet Computing*, **12(1)**, pp. 60–65.
- LOVE, P.E.; IRANIB, Z. y EDWARDS, D.J. (2004). «Industry-centric benchmarking of information technology benefits, costs and risks for small-to-medium sized enterprises in construction». *Automation in Construction*, **13(4)**, pp. 507–524.

- MACKENZIE, C.; LASKEY, K.; MCCABE, F.; BROWN, P. y METZ, R. (2006). «Reference Model for Service Oriented Architecture 1.0». *Informe tico*, OASIS.
- MADHUSUDANA, T. y YOUNG-JUN, S. (2005). «A simulation-based approach for dynamic process management at web service platforms». *Computers Industrial Engineering*, **49(2)**, pp. 287–317.
- MENASCÉ, D. (2002). «Load testing, benchmarking, and application performance Management for the Web». *Computer Measurement Group (CMG)*, pp. 1–11.
- MENASCÉ, D. (2003). «Scaling Web Sites Through Caching». *Internet Computing*, **7(4)**, pp. 86–89.
- MENASCÉ, D. y ALMEIDA, V. (1998). *Capacity Planning for Web Performance. Metrics, Models Methods*.
- MENASCÉ, D. y ALMEIDA, V. (2000). *Scaling for E-Business, Technologies, Models, Performance and Capacity Planning*.
- MENASCÉ, D.; ALMEIDA, V.; FONSECA, R. y MENDES, M. (2000). «Business-oriented resource management policies for e-commerce servers». *Performance Evaluation*, **42(2)**, pp. 223–239.
- MONTGÓMERY, D. C (2001). *Diseño y Análisis de Experimentos*.
- MULIK, S.; AJGAONKAR, S. y SHARMA, K. (2008). «Where Do You Want to Go in Your SOA Adoption Journey?» *IEEE Computer Society. IT Professional*, **10(3)**, pp. 36–39.
- NIDUMOLU, S.R.; ZEIGLER, B.P. y MENON, M.N. (1998). «Object-oriented business process modelling and simulation: A discrete event system especification framework». *Simulation Practice and Theory*, **6(6)**, pp. 533–571.

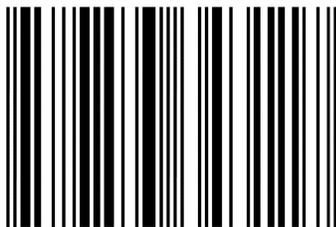
- NURCAN, S. (2008). «A Survey on the Flexibility Requirements Related to Business Processes and Modeling Artifacts». En: *In Proc. of the 41st Hawaii International Conference on System Sciences*, pp. 368–378. IEEE.
- OLIVER, T.; LEYKING, K. y DREIFUS, F. (2008). «Using Process Models for the Design of Service-Oriented Architectures: Methodology and E-Commerce Case Study». En: *In Proc. 41Hawaii International Conference on System Sciences*, .
- OSTERWALDER, A. y PIGNEUR, Y. (2002). «Organizational Characteristics and Information Systems: An Exploratory Investigation». *Academy of Management Journal*, **25(1)**, pp. 170–182.
- OSTERWALDER, A.; PIGNEUR, Y. y TRUCCI, C. L. (2005). «Clarifying business models: origins, present, and future of the concept». *Informe tico*, Association for Information Systems.
- PALANIAPPAN, S.; SAWHNEY., A. y SARJOUGHIAN, H. S. (2006). «Application of the DEVS Framework in Construction Simulation». En: *In Proc. of the 2006 Winter Simulation Conference*, pp. 2077–2086. IEEE.
- PAPAZOGLU, M.. (2003). «Service-Oriented Computing: Concepts, Characteristics and Directions». *Achieving supply chain excellence through technology*, **4**, pp. 42–43.
- PAPAZOGLU, M. y VAN DEN HEUVEL, W.-J. (2007). «Service oriented architectures: approaches, technologies and research issues». *European Journal of Operational Research*, **16(3)**, pp. 389–415.
- PFISTER, G. F. (1996). «Clusters of computers for commercial processing: the invisible architecture». *IEEE Parallel Distributed Technology: Systems Applications*, **4(3)**, pp. 12–14.

- PIRIS, L.; FITZGERALD, G. y SERRANO, A. (2004). «Strategic motivators and expected benefits from e-commerce in traditional organizations». *International Journal of Information Management*, **24(6)**, pp. 489–506.
- PUIGJANER, R.; SERRANO, J. y RUBIO, A. (1992). *Evaluación y Explotación de Sistemas Informáticos*. Ed. Síntesis.
- RANJAN, S. y KNIGHTLY, E. (2008). «High-Performance Resource Allocation and Request Redirection Algorithms for Web Clusters». *IEEE Transactions on Parallel and Distributed Systems*, **19(9)**, pp. 1186–1200.
- SARJOUGHIAN, HESSAM; KIM, SUNGUNG; RAMASWAMY, MUTHUKUMAR y YAU, STEPHEN (2008). «A simulation framework for service-oriented computing systems». En: *In Proc. of the 2008 Winter Simulation Conference*, pp. 845–853. IEEE Computer Society.
- SARJOUGHIAN, H.S.; ZEIGLER, B.P. y PARK, S. (2000). «Collaborative distributed network system: a lightweight middleware supporting collaborative DEVS modeling». *Future Generation Computer Systems*, **17(2)**, pp. 89–105.
- SCHMID, B. y LINDEMANN, M. (1998). «Elements of a Reference Model for Electronic Markets». En: *In Proc. of the 31st Hawaii International Conference on System Sciences*, pp. 193–201.
- SOLOMON, A. y LITOIU, M. (2011). «Business Process Performance Prediction on a Tracked Simulation Model». En: *In Proc. of the 3rd International Workshop on Principles of Engineering Service-Oriented Systems*, pp. 50–56. ACM.
- SOLOMON, A.; LITOIU, M.; BENAYON, J. y LAU, A. (2010). «Business Process Adap-

- tation on a Tracked Simulation Model». En: *In Proc. of the 2010 Conference of the Center for Advanced Studies on Collaborative Research*, pp. 184–198. ACM.
- SOTO ACOSTA, P. y MEROÑO CERDAN, A. (2008). «Analyzing e-business value creation from a resource-based perspective». *International Journal of Information Management*, **28(1)**, pp. 49–60.
- SOTO ACOSTA, P. y MEROÑO CERDAN, A. (2009). «Evaluating Internet Technologies Business Effectiveness». *Telematics and Informatics*, **26(2)**, pp. 211–221.
- SQUILLANTE, M. S.; XIA, C. H. y ZHANG, L. (2002). «Optimal scheduling in queuing network models of high-volume commercial web sites». *Performance Evaluation*, **47(4)**, pp. 223–242.
- STEPCHENKOVA, S.; TANG, L.; JANG, S.; KIRILENKO, A. y MORRISON, A. (2010). «Benchmarking CVB website performance: Spatial and structural patterns». *Tourism Management*, **31(5)**, pp. 611—620.
- SUBRAMANIAN, N. y RAMANATHAN, R. (2012). «A review of applications of Analytic Hierarchy Process in operations management». *International Journal of Production Economics*.
- TAHA, H. A. (2004). *Investigación de Operaciones. 7a. Edición*. Ed. Pearson Education. ISBN 970-26-0498-2.
- TSAI, Y.CH. y CHENG, YT. (2011). «Analyzing key performance indicators (KPIs) for E-commerce and Internet marketing of elderly products: A review». *Archives of Gerontology and Geriatrics*.
- VAN DER AALST W. M. P. (2006). «Pi calculus versus Petri nets: Let us eat humble pie

- rather than further inflate the Pi hype». En: *In Proc. of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*, volumen 3, pp. 1–11.
- WAINER, G. (2003). *Metodologías de Modelización y Simulación de Eventos Discretos*. Ed. Nueva Librería.
- WANG, H.; HUANGC, J. Z.; QUB, Y. y XIE, J. (2004). «Web Services: Problems and Future Directions». *Web Semantics: Science, Services and Agents on the World Wide Web*, **1(3)**, pp. 309–320.
- XIAOPING, XU y CHEN, LI (2008). «The Supply Chain Performance Evaluations Indicator System Based on Benchmark Balanced Scorecard». En: *In Proc. of the 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4. IEEE.
- ZEIGLER, B. (1976). *Theory of Modelling and simulation*. Wiley. ISBN 4-71981-52-4.
- ZEIGLER, B.P. (2003). «DEVS Today: Recent Advance in Discrete Event-Based Information Technology». En: *In Proc. 11th IEEE/ACM International Symposium on Modeling Analysis and Simulation of Computer Telecommunications Systems*, IEEE Computer Society/ACM.
- ZEIGLER, B.P. y SARJOUGHIAN, H. S. (2003). «Introduction to DEVS Modelling Simulation with JAVA: Developing Component-Based Simulation Models». *Informetico*, University of Arizona.
- ZHANG, Z. y FAN, W. (2008). «Web server load balancing: A queueing analysis». *European Journal of Operational Research*, **186(1)**, p. 681–693.
- ZU EISSEN, S. M. y STEIN, B. (2006). «Realization of Web-based simulation services». *Computers in Industry*, **57**, pp. 261–271.

ISBN 978-987-33-4311-7



9 789873 343117