

**COORDINACION DE MODELOS MATEMATICOS,  
USANDO BLUEJADE  
(INTEGRACION JBOSS, JADE, JBPM)**

**CARLOS RENE SERRATO ENRIQUEZ**

**FUNDACION UNIVERSITARIA SAN MARTIN  
FACULTAD DE INGENIERIA  
PROGRAMA DE INGENIERIA DE SISTEMAS  
BOGOTA, D.C.  
2010 I**

Ti

R# 3104

**COORDINACIÓN DE MODELOS MATEMÁTICOS, USANDO BLUEJADE  
(INTEGRACIÓN JBOSS, JADE, JBPM)**

**CARLOS RENÉ SERRATO ENRÍQUEZ**

**FUNDACIÓN UNIVERSITARIA SAN MARTÍN  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
BOGOTÁ  
2010 I**

**COORDINACIÓN DE MODELOS MATEMÁTICOS, USANDO BLUEJADE  
(INTEGRACIÓN JBOSS, JADE, JBPM)**

**CARLOS RENÉ SERRATO ENRÍQUEZ  
031115  
CR.SERRATO@GMAIL.COM**

**MONOGRAFÍA DE GRADO**

**ASESOR TÉCNICO  
FABIAN ANDRES GIRALDO GIRALDO  
GRUPO DE INTERES EN CONSTRUCCION DE SOFTWARE**

**FUNDACIÓN UNIVERSITARIA SAN MARTÍN  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA DE SISTEMAS  
BOGOTÁ  
2010 I**

**Nota de aceptación**

---

---

---

---

---

---

---

**Fabián Andrés Giraldo Giraldo**  
**Asesor**

---

**Diego Giraldo**  
**Jurado**

---

**Adriana Guerrero**  
**Jurado**

Bogotá, 04 de Junio 2010

A Dios por darme la oportunidad,  
fuerza y ganas para salir adelante.

A mi familia, amigos y a cada persona  
allegada a mi, que con su paciencia,  
amor, conocimiento, amistad y apoyo  
me impulsaron a no darme por vencido  
y superar cada obstáculo en el camino.

*"Si te caes siete veces, levántate ocho."*

*Proverbio Chino*

## AGRADECIMIENTOS

Agradezco a Dios por brindarme sabiduría para sortear todos los baches en el camino, la fortaleza para cuando caí levantarme y las ganas para cuando me desanime salir adelante.

A mis padres Luis Carlos y Yuri, que con su apoyo incondicional, su inmenso amor, sus manos siempre ahí para brindarme todo lo que necesité y su gran ejemplo, me recordaron y me recordarán que para todo hay un sacrificio pero que cuando se consigue lo que se quiere, no hay nada más gratificante que el deber cumplido.

A mi hermana que siempre ha sido una amiga, una confidente y la mejor de las hermanas.

A mis abuelas, que nunca dejaron de cuidarme y guiarme, las cuales siempre quisieron ver el día en que yo lograra este objetivo y a pesar que no estén conmigo se que desde algún lugar están disfrutando tanto como yo que todo se hizo una realidad.

A Paola que estuvo ahí apoyándome en todo, siempre tuvo una palabra de ánimo cuando más la necesité y su cariño no dejó que me rindiera.

A mis grandes amigos Mauricio, Juan Felipe, Christian, Juan Pablo, Jorge A., Jeisson, Jorge, German, Jhonatan, Pupo, Gina, que siempre estuvieron ahí para apoyarme, regañarme, aconsejarme y sobretodo animarme para cumplir un gran objetivo.

Y finalmente a mi asesor Fabián Andrés Giraldo, con sus grandes ideas, su increíble conocimiento y su incondicional compromiso con el proyecto, hizo que todo esto fuera posible, no solo como asesor sino como amigo, siempre me impulsó a hacer más y a entregar lo mejor de mí como persona y estudiante.

## CONTENIDO

|   |    |
|---|----|
| INTRODUCCIÓN.....   | 16 |
| 1. PROBLEMA.....  | 17 |
| 2. JUSTIFICACIÓN.....   | 19 |
| 3. OBJETIVOS.....   | 21 |
| 3.1 OBJETIVO GENERAL.....   | 20 |
| 3.2 OBJETIVOS ESPECÍFICOS .....   | 20 |
| 4.1 ANTECEDENTES .....  | 21 |
| 4.1.1 Distributed Workflow Enactment: an Agent-based Framework (Fortino, 2006)  | 21 |
| 4.1.2 A Workflow and Agent based Platform for Service Provisioning (Shrivastava ., 2000)                                  | 21 |
| 4.1.3 Automating distributed workflow for electronic commerce: A model for building meta-workflow components (Gill, 1999) | 22 |
| 4.2 MARCO CONCEPTUAL.....   | 23 |
| 4.2.1 Modelo Matemático (Vignolo, 2005).....  | 23 |
| 4.2.2 Componentes de los modelos de programación lineal (Linares, 2001)....   | 25 |
| 4.2.3 Modelos matemáticos de optimización (Baíllo, 2004) .....  | 26 |
| 4.2.4 Tipos de lenguajes de modelamiento matemático (Vignolo, 2005) .....   | 27 |
| 4.3 MARCO TEÓRICO.....  | 29 |
| 4.3.1 Los Sistemas Multi-Agentes (Sierra, 1997), (Vicente, 2003).....   | 29 |
| 4.3.2 JADE Agent Development (Alonso, 2001), ("JADE", 2009) .....   | 35 |

|       |   |     |
|-------|---|-----|
| 4.3.3 | J2EE ("Java Enterprise Edition", 2009) .....                          | 39  |
| 4.3.4 | JBoss ("Red Hat" 2009).....   | 43  |
| 4.3.5 | BlueJADE (Nilsen, 2005).....  | 46  |
| 4.3.6 | JBPM ("jboss.org: community driven", 2009) .....                      | 53  |
| 4.3.7 | FreeMarker ("FreeMarker", 2009) .....                                 | 57  |
| 4.3.8 | Velocity (Naccarato, 2004).....                                       | 64  |
| 4.4   | ESTADO DEL ARTE .....   | 66  |
| 4.4.1 | Managing Software Agents In J2EE Application Servers (Nilsen, 2005).. | 66  |
| 4.5   | LIMITACIONES Y ALCANCES.....  | 69  |
| 4.5.1 | División del problema en los modelos que lo conforman .....           | 69  |
| 4.5.2 | Fuentes de Datos .....  | 69  |
| 4.5.3 | Generación de Flujo de Trabajo (Workflow).....                        | 69  |
| 4.5.4 | Adición Funcionalidades .....   | 69  |
| 5.1   | OPENUP (Open UP, 2008) .....  | 70  |
| 5.1.1 | Roles en la metodología OpenUP (Kroll, 2009) .....                    | 80  |
| 5.2   | aplicación de la Metodología al Proyecto .....                        | 81  |
| 5.2.1 | Fase de Elaboración: .....  | 81  |
| 5.2.2 | Fase de Construcción: .....   | 82  |
| 5.2.3 | Fase de Transición:.....  | 87  |
| 6.1   | Introducción.....   | 118 |

|       |  |     |
|-------|--|-----|
| 6.2   | Creación .....   | 118 |
| 6.2.1 | Inicio del Proyecto .....  | 119 |
| 6.2.2 | Identificar y Afinar Requerimientos .....  | 119 |
| 6.2.3 | Planificar y Gestionar iteración.....  | 120 |
| 6.3   | Elaboración.....   | 120 |
| 6.3.1 | Desarrollo de la Arquitectura.....   | 121 |
| 6.3.2 | Especificación casos de uso .....  | 124 |
| 6.3.3 | Plan de Pruebas.....   | 125 |
| 6.3.4 | Riesgos .....  | 126 |
| 6.4   | Construcción.....  | 126 |
| 6.4.1 | Iteración I Generar plantillas Velocity para modelos matemáticos Java-Cplex. 127   |     |
| 6.4.2 | Iteración II Coordinar modelos matemáticos basándose en Workflow JBPM 144  |     |
| 6.4.3 | Iteración III Generar plantillas FreeMarker para modelos matemáticos JADE- BlueJADE.....                                 | 162 |
| 6.4.4 | Iteración IV Generar plantillas FreeMarker que permitan coordinar modelos matemáticos en BlueJADE a través de JBPM ..... | 173 |
| 6.4.5 | Iteración V Generación de escenario de prueba para coordinación de agentes que ejecuten modelos matemáticos. ....        | 175 |
| 6.5   | Transición.....  | 186 |
| 6.5.1 | Integración .....  | 186 |

|       |                                       |     |
|-------|---------------------------------------|-----|
| 7.1   | Pruebas .....                         | 187 |
| 7.1.1 | Documentos prueba Iteración I .....   | 187 |
| 7.1.2 | Documentos prueba Iteración II .....  | 187 |
| 7.1.3 | Documentos prueba Iteración III ..... | 187 |
| 7.1.4 | Documentos prueba Iteración IV .....  | 187 |
| 7.1.5 | Documentos prueba Iteración V .....   | 187 |
| 8.    | CONCLUSIONES .....                    | 150 |
| 9.    | TRABAJO FUTURO .....                  | 152 |
|       | GLOSARIO .....                        | 153 |
|       | BIBLIOGRAFÍA .....                    | 156 |

## LISTA DE TABLAS

|   | pág. |
|---|------|
| Tabla 1. Tipos de problemas LP según su tamaño  | 24   |
| Tabla 2. Componentes de los Modelos de Programación Lineal  | 25   |
| Tabla 3. Documentación del proyecto.  | 81   |
| Tabla 4. Iteración 1, generación de plantillas Velocity para modelos Java-CPLEX.                    | 82   |
| Tabla 5. Iteración 2, coordinación de modelo mediante JBPM.   | 83   |
| Tabla 6. Iteración 3, generación de plantillas FreeMarker para modelos JADE-BlueJADE.               | 84   |
| Tabla 7. Iteración 4, generación de plantillas FreeMarker para coordinación de modelos en BlueJADE. | 85   |
| Tabla 8. Iteración 5, escenario de prueba.  | 86   |
| Tabla 9. Documentación  | 87   |

## LISTA DE FIGURAS

|  | pág. |
|--|------|
| Figura 1. Estructura básica de JADE                                      | 35   |
| Figura 2. Comportamientos y protocolos FIPA.                             | 39   |
| Figura 3. Modelo de aplicación empresarial                               | 40   |
| Figura 4. Arquitectura plataforma J2EE                                   | 42   |
| Figura 5. Arquitectura JBoss   | 43   |
| Figura 6. Segmento de código búsqueda lookup                             | 48   |
| Figura 7. Integración BlueJADE con JBoss                                 | 49   |
| Figura 8. BlueJADE acoplado al servidor JBoss                            | 49   |
| Figura 9. Archivo jade-service.xml                                       | 51   |
| Figura 10. Continuación del archivo jade-service.xml                     | 51   |
| Figura 11. Archivo jade-service-config.xml                               | 52   |
| Figura 12. Ejemplo de configuración de jadeboot.properties               | 53   |
| Figura 13. Proceso de trabajo de FreeMarker                              | 58   |
| Figura 14. Estructura Plantilla FreeMarker                               | 59   |
| Figura 15. Ejemplo de modelo de datos                                    | 60   |
| Figura 16. Ejemplo sequence en modelos de datos.                         | 62   |
| Figura 17. Arquitectura Velocity   | 64   |
| Figura 18. Modelo de Objeto Interior                                     | 65   |
| Figura 19. Plataforma de Modelo Específico                               | 66   |
| Figura 20. BlueJADE acoplado a servidor JBoss                            | 67   |
| Figura 21. BlueJADE Web manager: Interfaz de Despliegue.                 | 68   |
| Figura 22. BlueJADE Web manager: Agentes ejecutándose en una plataforma. | 68   |
| Figura 23. Capas de la metodología OpenUP                                | 72   |
| Figura 24. Fases del Ciclo de Vida de un Proyecto                        | 74   |
| Figura 25. Fase de Inicialización  | 75   |
| Figura 26. Fase de Elaboración   | 76   |
| Figura 27. Fase de Construcción  | 76   |
| Figura 28. Fase de Transición  | 77   |

|   |     |
|---|-----|
| Figura 29. Reducción de Riesgo (curva roja) y Creación de Valor (curva verde) Durante el Ciclo de Vida del Proyecto | 78  |
| Figura 30. Una Iteración Pasa por un Ciclo de Vida  | 79  |
| Figura 31. Roles de la Metodología OpenUP   | 80  |
| Figura 32. Actividades a realizar en la Fase de Creación.   | 118 |
| Figura 33. Actividades a realizar en la Fase de Elaboración   | 121 |
| Figura 34. Metamodelo Final   | 123 |
| Figura 35. Actividades a realizar en la Fase de Construcción  | 126 |
| Figura 36. Diagrama de Clases Metamodelo  | 128 |
| Figura 37. Diagrama de paquetes   | 132 |
| Figura 38. Diagrama final de clases   | 134 |
| Figura 39. Archivo Problema.hbm   | 136 |
| Figura 40. Archivo Problema Vo  | 137 |
| Figura 41. Segmento Archivo Facade  | 138 |
| Figura 42. Segmento Archivo ProblemaDao   | 139 |
| Figura 43. Archivo generado CPLEX   | 140 |
| Figura 44. Plantilla Velocity para generación modelo Java-Cplex   | 143 |
| Figura 45. Definición y representación gráfica de secuencia   | 145 |
| Figura 46. Definición y representación en código de secuencia   | 146 |
| Figura 47. Clase modelo1ActionHandler   | 147 |
| Figura 48. Definición y representación gráfica de decisión  | 148 |
| Figura 49. Definición y representación en código de decisión  | 149 |
| Figura 50. Segmento de código específico de decisión  | 149 |
| Figura 51. Definición y representación gráfica de concurrencia  | 150 |
| Figura 52. Definición y representación en código de concurrencia  | 151 |
| Figura 53. Definición y representación gráfica de iteración   | 153 |
| Figura 54. Definición y representación en código de iteración   | 154 |
| Figura 55. Modelo Entidad - Relación  | 155 |
| Figura 56. Interfaz Configuración Nodos Workflow  | 155 |
| Figura 57. Configuración hibernate.cfg.xml  | 156 |
| Figura 58. Archivo VO nodo  | 157 |
| Figura 59. Archivo VO transicion  | 158 |

|  |     |
|--|-----|
| Figura 60. Interfaz Configuración Transiciones Workflow                  | 159 |
| Figura 61. Segmento de código creación hashmap                           | 160 |
| Figura 62. Segmento código plantilla FreeMarker                          | 161 |
| Figura 63. Segmento código plantilla Agente Java-CPLEX                   | 163 |
| Figura 64. Segmento código agente generado Java-cplex                    | 164 |
| Figura 65. segmento de código plantilla agente Java-CPLEX (2)            | 165 |
| Figura 66. Segmento código generado de Agente Java-CPLEX (2)             | 165 |
| Figura 67. Archivo jade-service.xml                                      | 166 |
| Figura 68. plantilla FreeMarker jade-service.ftl                         | 167 |
| Figura 69. Segmento de código hasmap jade-service                        | 167 |
| Figura 70. archivo jade-service-config.xml                               | 168 |
| Figura 71. Plantilla FreeMarker jade-service-config.ftl                  | 169 |
| Figura 72. Segmento de código hashmap jade-service-config                | 170 |
| Figura 73. Archivo jadeboot.properties                                   | 171 |
| Figura 74. Estructura Plantilla FreeMarker creación application.jar      | 171 |
| Figura 75. Estructura de carpetas para creación de grupo BlueJADE        | 173 |
| Figura 76. Estructura archivo install.bat del grupo generado en BlueJADE | 174 |
| Figura 77. Estructura carpetas de grupo generado en BlueJADE             | 176 |
| Figura 78. Estructura carpetas generada para equipos en la red           | 177 |
| Figura 79. Estructura carpetas generada para Workflow                    | 177 |
| Figura 80. Estructura carpetas generada para Equipos                     | 179 |
| Figura 81. Segmento de archivo processdefinition.xml                     | 180 |
| Figura 82. Consola JBPM  | 182 |
| Figura 83. Iniciación del workflow                                       | 183 |
| Figura 84. Worflow en ejecución  | 184 |
| Figura 85. Worflow en ejecución 2  | 185 |
| Figura 86. Actividades a realizar en la Fase de Transición               | 186 |

## RESUMEN

El proyecto de grado, "Coordinación de modelos matemáticos usando BlueJADE (Integración JBoss, JADE, JBPM)", es una aplicación que provee a los modeladores matemáticos procesos y mecanismos para la ejecución cooperativa de modelos matemáticos complejos, siendo capaz de orquestar un problema, brindando una mejora significativa en los tiempos de procesamiento y ejecución de estos modelos de optimización lineal.

La herramienta no solo provee gran productividad al proporcionar la orquestación de los modelos, sino al brindar la generación de código en menos tiempo de lo que demoraría un programador experto, si no que asegura que todo lo generado funcione correctamente y quede listo para ser ejecutado. El código que genera la herramienta es en lenguaje de propósito general Java enlazado con el solver de optimización CPLEX, el cual contiene un conjunto de librerías con las que se realiza todo el procesamiento matemático del problema.

Para el desarrollo del proyecto se elaboro toda la planeación con la metodología OpenUP que divide todo el proceso en fases y estas a su vez en iteraciones las cuales cuentan con micro incrementos, lo que permite validar en cada iteración el avance, facilitando así el seguimiento y control.

## INTRODUCCIÓN

En la formulación del modelo matemático se usan diferentes lenguajes algebraicos de modelamiento que permiten encontrar la solución al modelo de una forma rápida y confiable. Actualmente existe un gran número de lenguajes algebraicos de modelamiento, estos se diferencian en costos, sintaxis y restricciones tecnológicas debido a que cada organización utiliza el lenguaje que se acomode a sus necesidades.

Esta diversidad de lenguajes hace que los usuarios tengan que destinar gran parte del tiempo a aprender estos lenguajes, a tener destrezas en el manejo de un lenguaje de propósito general y a poder enlazarlo con un solver de optimización para potenciar más el modelo diseñado; este tiempo que se emplea, se puede invertir en otras fases del desarrollo de investigación de operaciones. Adicionalmente gran parte de estos modelos tienden a ser complejos y su nivel de procesamiento poco eficiente por la cantidad de variables y datos que estos contienen.

Por esta razón se hace necesario un sistema de coordinación de modelos basado en agentes y técnicas de Workflow que permita independizar el proceso de ejecución de los diferentes modelos matemáticos y así mismo la generación automática de código en un lenguaje de propósito general enlazado con un solver de optimización.

Como solución a estas problemáticas se plantea una aplicación que permita realizar las siguientes funcionalidades:

- Generación automática del código en el lenguaje de propósito JAVA general enlazado con el solver de optimización CPLEX, el código es cien por ciento ejecutable.
- Implementación de procesos y mecanismos de coordinación de agentes para la ejecución de estos modelos matemáticos complejos, esto con el fin de descentralizar el procesamiento de estos en una sola máquina y brindando la oportunidad de optimizar el tiempo de ejecución de estos grandes modelos.