

Tendencias Tecnológicas en Computación

Research in Computing Science

Series Editorial Board

Comité Editorial de la Serie

Editors-in-Chief:

Editores en Jefe

Grigori Sidorov (Mexico)
Gerhard Ritter (USA)
Jean Serra (France)
Ulises Cortés (Spain)

Associate Editors:

Editores Asociados

Jesús Angulo (France)
Jihad El-Sana (Israel)
Jesús Figueroa (Mexico)
Alexander Gelbukh (Russia)
Ioannis Kakadiaris (USA)
Serguei Levachkine (Russia)
Petros Maragos (Greece)
Julian Padget (UK)
Mateo Valero (Spain)

Editorial Coordination:

Coordinación Editorial

Socorro Méndez Lemus

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 66**, octubre 2013. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título No. : 04-2005-121511550100-102*, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título No. 12897*, *Certificado de licitud de Contenido No. 10470*, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 66**, Oct 2013. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 66

Volumen 66

Tendencias Tecnológicas en Computación

Editores del volumen:

Volume Editors

Juan Carlos Herrera Lozada

Magdalena Marciano Melchor

Hind Taud

Grigori Sidorov

Miguel Gabriel Villarreal Cervantes

Instituto Politécnico Nacional
Centro de Investigación en
Computación
México 2013



Instituto Politécnico Nacional
"La Técnica al Servicio de la Patria"



ISSN: 1870-4069

Copyright © Instituto Politécnico Nacional 2013

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

<http://www.ipn.mx>

<http://www.cic.ipn.mx>

The editors and the publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

All rights reserved. No part of this publication may be reproduced, stored on a retrieval system or transmitted, in any form or by any means, including electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the Instituto Politécnico Nacional, except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper.

Indexed in LATINDEX and Periodica / Indexada en LATINDEX y Periódica

Printing: 500 / Tiraje: 500

Printed in Mexico / Impreso en México

Table of Contents

Page

Interdependencias del Sistema de Transporte Colectivo Metro y la Red de Transporte de Pasajeros con autómatas celulares	03
<i>Diego Alfredo Padilla Pérez, Jaime Reynaldo Santos Reyes, Luis Manuel Hernández Simón, José Luis González Domínguez, Tatiana Gouseva, Galdino Cardenal Santos Reyes, Verónica Mendoza González</i>	
Identificación de los principios de segmentación y extracción de TRIZ en patrones de diseño	17
<i>María Erika Auryly García-Cantú, Ulises Juárez-Martínez, Guillermo Cortés-Robles</i>	
Bon Iterative Graph Feature Mining for Graph Indexing	55
<i>A.Pankaj Moses Monickaraj, K. Vivekanandan, D. Ramya Chitra</i>	
Desarrollo de una aplicación de Software para simulación de maquinado en un torno básico CNC para el Instituto Tecnológico de Orizaba	65
<i>Héctor Uriel Osorio Dávila, Hilarión Muñoz Contreras</i>	
Aplicaciones colaborativas con realidad aumentada para dispositivos móviles basados en Android	73
<i>Joaquín Llanillo Barquet, María Antonieta Abud Figueroa, Ignacio López Martínez, Celia Romero Torres</i>	
Aplicación de un navegador 3D con una interfaz de usuario natural utilizando la plataforma .NET, SDK y Kinect de Microsoft.....	83
<i>Peredo Valderrama, K. Anaya Rivera, R. Peredo Valderrama</i>	
Administración de procesos de reingeniería de software orientada a aspectos.....	95
<i>Isabel Espinoza-Espinoza, Ulises Juárez-Martínez</i>	
Factor de Impacto sobre artículos: propuesta de indicadores que obtengan el aporte de los artículos individuales en la comunidad científica.....	109
<i>Jesús Vázquez-Ruiz, Beatriz Alejandra Olivares-Zepahua, María Antonieta Abud-Figueroa</i>	

Análisis comparativo de marcos de trabajo para el desarrollo de servicios (Cloud Services) y aplicaciones en la nube (Cloud Computing)	137
<i>Rafael Antonio Fernández Domínguez, Giner Alor Hernández</i>	
Desarrollo de una herramienta Web para monitorización y control de proyectos de software para empresas con nivel 5 de CMMI-DEV	151
<i>D. A. Ramos Calvario, A. M. Chávez Trejo, Z.B. Osorio Martínez</i>	
Security in Web applications	161
<i>A. Santoyo-Sanchez, C. De Jesús-Velásquez, L. I. Aguirre-Salas</i>	

Interdependencias del Sistema de Transporte Colectivo Metro y la Red de Transporte de Pasajeros con autómatas celulares

Diego Alfredo Padilla Pérez^{1,2}, Jaime Reynaldo Santos Reyes^{1,2},
Luis Manuel Hernández Simón¹, José Luis González Domínguez¹,
Tatiana Gouseva^{1,2}, Galdino Cardenal Santos Reyes^{1,2}, Verónica Mendoza González¹

¹Instituto Politécnico Nacional, Sección de Estudios de Posgrado e Investigación.

²Grupo de Investigación SARACS (Safety, Accident, Risk & Reliability Analysis)

dapswy27_ipn@hotmail.com

Resumen. Las relaciones de las infraestructuras del Sistema de Transporte Colectivo Metro con la Red de Transporte de Pasajeros, en la Ciudad de México, se llaman interdependencias. Estas relaciones son bidireccionales, es decir, el estado de cada infraestructura se correlaciona con el estado de la otra infraestructura y viceversa. En este trabajo modelamos las interacciones e interdependencias Metro con la Red de Transporte de Pasajeros (RTP) cuando el primero no está disponible. El modelado de las interdependencias se realizó con la simulación de autómatas celulares en Matlab. Las interdependencias entre estos dos sistemas se representan de la siguiente manera: interdependencias fuertes están representadas en color rojo, una interdependencia débil en color anaranjado y si no hay interdependencias se utiliza el color negro. Con este modelo se puede representar diferentes escenarios entre la falta de disponibilidad del Metro y la canalización de los pasajeros hacia la Red de Transporte.

Palabras clave: Interdependencias, Infraestructuras, Metro, Autómatas Celulares.

1. Introducción

Las infraestructuras se encuentran conectadas en múltiples puntos, de tal manera que existe una relación bidireccional entre los estados de dos infraestructuras determinadas. En otras palabras, sea la infraestructura 1 la cual depende de la infraestructura 2 a través de la relación i y viceversa, la infraestructura 2 que depende de la infraestructura 1 a través de la relación j ; adicionalmente, ambas infraestructuras depende de la relación $i j$.

Los Tipos de Interdependencias tienen características distintas, estas clases de interdependencias no son y no deben de ser mutuamente excluyentes. Los tipos de interdependencias son: física, cibernética, geográfica y lógica [1].

Interdependencia física se tiene cuando el estado de una infraestructura depende de la producción de material de la otra infraestructura. Esto se genera entre la vinculación de entradas y salidas.

Interdependencia cibernética se genera en el momento en que una infraestructura requiere de información en enlaces electrónicos de otra infraestructura.

Interdependencia geográfica se produce en el momento en que dos o más infraestructuras se encuentran en estrecha proximidad espacial.

Dos infraestructuras son lógicamente interdependientes si el estado de cada una depende del estado de la otra a través de la validez de los argumentos en términos de su estructura.

Una cosmovisión de las interdependencias ha demostrado los peligros inherentes que se encuentran en las propiedades emergentes. Por tal motivo, la Comisión Presidencial de los EE. UU., señaló que "no hay prioridad más urgente que garantizar la seguridad, continuidad y disponibilidad de las infraestructuras críticas" [2]

El informe de la Comisión Presidencial de EE.UU., sobre protección de infraestructuras críticas articuló los peligros inherentes de las interdependencias no controladas, pero no proporcionó una metodología para pensar o analizar este fenómeno [3].

Los autómatas celulares (CA) son algunos de los modelos más antiguos de la computación natural, que data de más de medio siglo. Los primeros estudios de CA por John von Neumann en la década de 1940 fueron biológicamente motivados [4][5]; el objetivo era diseñar sistemas artificiales autor-replicantes que también son computacionalmente universales. Von Neumann define un universo discreto que consta de una malla bidimensional de máquinas de estados finitos, llamadas células, interconectadas entre sí a nivel local. Las células cambian de forma sincrónica estados en función de los estados de algunas células cercanas, los vecinos, según lo determinado por una regla de actualización local.

Todas las células utilizan la misma regla de actualización para que el sistema es homogéneo al igual que muchos sistemas físicos y biológicos. Estos universos celulares son conocidos como Autómata Celular [6].

Los estados del autómata provienen de un estado conjunto finito S . En un momento dado, la configuración del autómata es una aplicación $c: Z^d \rightarrow S$ que especifica los estados de todas las células. El conjunto S^{Z^d} de todas las configuraciones se denota por $C(d, S)$, o brevemente C cuando d y S son conocidos a partir del contexto [7].

Las CA, son infinitos conjuntos de máquinas de estados finitos llamadas células e indexado por Z . Estas células evolucionan sincrónicamente en pasos de tiempo discretos siguiendo una regla local por el cual se determina el estado de una célula en función de su propio estado, junto con los estados de sus dos vecinos [8].

2. Modelado de las Interdependencias

El modelado de las interdependencias está constituido por el método sistémico del proceso cibernético participativo de investigación–acción y los pasos de diagnóstico, diseño y acción. El diagnóstico tiene la información de los eventos externos que pueden afectar a la infraestructura y cómo la afectan. Esta información define las entidades involucradas.

El diseño está compuesto por cinco etapas que son: entidades involucradas, dirección de la relación, tipos de Interdependencia, estado de relación y riesgo.

La naturaleza de las entidades involucradas generan las dependencias de instancias de personas–personas, personas–material, personas–instalaciones y material – instalaciones. La dirección de la relación entre las entidades involucradas y el tipo de interdependencia pueden ser unidireccionales o bidireccionales.

La compartición entre las entidades involucradas definen los tipos de interdependencia que son: Física (un elemento producido por uno depende del consumido del otro elemento), Cibernética (flujo de información electrónica), geográfica (se produce en el momento en que dos o más elementos se encuentran en estrecha proximidad espacial), lógica (el estado del elemento depende a través de la validez de otro elemento).

La etapa del estado de relación son los eventos que afectan a las entidades involucradas. Los estados de relación están expuestos a amenazas y tienen vulnerabilidades. Si las amenazas explotan las vulnerabilidades generan un riesgo que ocasiona un impacto.

Las interdependencias están determinadas con base en su impacto en la siguiente infraestructura. El impacto está determinado de la siguiente forma:

- Interdependencia Alta
- Interdependencia Media
- Interdependencia Baja

2.1 Interdependencias del Metro

Para analizar las interdependencias del Metro se correlacionará con los usuarios, trenes, instalaciones, conductores, mantenimiento y central de control [9] [10] [11]

En la Tabla 1, Tabla 2 y Tabla 3 se muestra los tipos de entidades involucradas, tipo de interdependencia y los estados de relación del sistema de transporte colectivo Metro.

Tabla 1. Interdependencias del Metro (Usuarios)

<i>Entidades</i>	<i>Interdependencia</i>	<i>Estado</i>
Usuarios – Trenes	Lógica	1.- Horas pico y Eventos Masivos
		2.- Vendedores 3.-Actos vandálicos
Usuarios – Instalaciones	Geográfica	4.- Aglomeracion y Violencia
		5.- Basura
		6.-Escaleras Eléctricas

Tabla 2. Interdependencia del Metro (Infraestructura).

<i>Entidades</i>	<i>Interdependencia</i>	<i>Estado</i>
Trenes – Mantenimiento	Física	7.- Suministro de Energía Eléctrica
	Lógica	8.- Falta de Mantenimiento
Mantenimiento – Instalaciones	Geográfica	9.- Lluvia 10.- Sismos
	Lógica	11.- Terrorismo
Instalaciones – Central de Control	Cibernética	12.-Información de Reportes
Central de Control – RTP	Cibernética	13.Información de Reportes

Tabla 2. Interdependencias del Metro (Conductores).

<i>Entidades</i>	<i>Interdependencia</i>	<i>Estado</i>
Trenes – Instalaciones	Lógica	14.- Retrasos en los Trenes
	Geográfica	15.- Vías
Trenes – Conductores	Lógica	16.- Ausencias
		17.- Estado Inconveniente

2.2. Interdependencias de la Red de Transporte de Pasajeros

Para analizar las interdependencias de la Red de Transporte de Pasajeros se correlacionará con los usuarios, camiones RTP, conductores, mantenimiento y módulos. [12]. En la Tabla 4 se tienen los tipos de entidades involucradas, tipo de interdependencia y los estados de relación de la Red de Transporte de Pasajeros.

Tabla 3. Interdependencias de la Red de Transporte de Pasajeros.

<i>Entidades</i>	<i>Interdependencia</i>	<i>Estado</i>
Usuarios – Camiones	Lógica	1.- Horas Pico, Eventos Masivos, Tráfico y Manifestaciones
		2.- Actos Vandálicos
Camiones – Conductores	Física	3.- Ausencias
		4.- Estado Inconveniente
Camiones – Mantenimiento	Lógica	5.- Falta de Mantenimiento
Camiones – Módulo Operativo	Lógica	6.- Ubicación de los Módulos
		7.- Cierre de los Módulos

3. Desarrollo

En 1984 Wolfram propuso una clasificación en la que se recogen aquellas situaciones o estados a los que un autómata celular puede evolucionar tras un cierto tiempo de simulación, número de iteraciones o ciclos [13]. Para realizar las interdependencias con el modelo de autómatas celular entre el Metro y la Red de Transporte de Pasajeros se realiza una matriz de transición entre cada entidad. La matriz de transición se obtiene del diagrama de estados de las entidades. En la Figura 1 se muestra el diagrama de estados de las entidades involucradas (Tabla 1, Tabla 2 y Tabla 3) en el Metro.

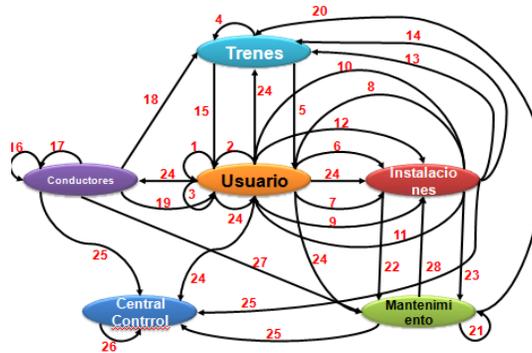


Fig. 1. Diagrama de Estados de las Entidades en el Metro.

Con base en el diagrama de estados se realiza la matriz de transición. En la Tabla 5 se observa la matriz de transición de cada estado de las entidades en el metro.

Table 4. Matriz de Transición del Metro.

	<i>Usuario</i>	<i>Trenes</i>	<i>Instalaciones</i>	<i>Conductores</i>	<i>Mantenimiento</i>	<i>Central Control</i>
Usuario	$12x_1$	$2x_2$	$3x_3$	x_4	0	0
Trenes	x_1	$3x_2$	$2x_3$	x_4	x_5	0
Instalaciones	$5x_1$	0	$8x_3$	0	x_5	0
Conductores	x_1	0	0	$6x_4$	0	0
Mantenimiento	x_1	0	$2x_3$	x_4	$4x_5$	0
Central Control	x_1	0	x_3	x_4	x_5	x_6

En la figura 2 se tiene el diagrama de estados de las entidades involucradas (Tabla 4) en la Red de Transporte de Pasajeros.

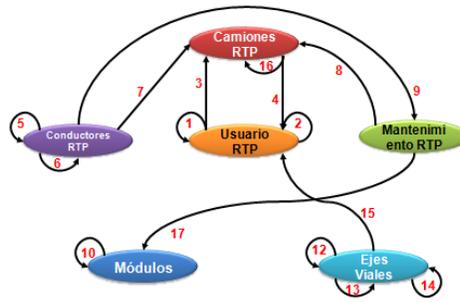


Fig. 2. Diagrama de Estados de las Entidades en la RTP.

En la Figura 2 se le ha agregado una entidad más que es ejes viales, debido a que este sistema hace uso de esta entidad. Con base en el diagrama de estados se realiza la matriz de transición. En la Tabla 6 se muestra la matriz de transición de cada estado de las entidades en la red de transporte de pasajeros.

Tabla 5. Matriz de Transición de la RTP.

	<i>Usuarios</i>	<i>Camiones</i>	<i>Mantenimiento</i>	<i>Conductores</i>	<i>Módulos</i>	<i>Ejes Viales</i>
Usuarios	$3x_7$	x_8	0	0	0	0
Camiones	x_7	$2x_8$	x_9	x_{10}	x_{11}	x_{12}
Mantenimiento	0	0	$2x_9$	x_{10}	0	0
Conductores	0	0	0	$4x_{10}$	0	0
Módulos	0	0	x_9	0	$3x_{11}$	0
Ejes Viales	0	0	0	0	0	$4x_{12}$

Para generar las interdependencias entre Metro y la Red de Transporte de Pasajeros, se juntaran las dos matrices de transición en forma diagonal y la intersección entre cada entidad serán las interdependencias. En la Figura 3 se observa la matriz de interdependencias.

	usuario	Trenes	Instalaciones	Conductores	Mantenimiento	Central Control	Usuarios	Camiones	Mantenimiento	Conductores	Módulos	Ejes Viales
Usuario	$12x_1$	$2x_2$	$3x_3$	x_4	0	0	Interdependencias					
Trenes	x_1	$3x_2$	$2x_3$	x_4	x_5	0						
Instalaciones	$5x_1$	0	$8x_3$	0	x_5	0						
Conductores	x_1	0	0	$6x_4$	0	0						
Mantenimiento	x_1	0	$2x_3$	x_4	$4x_5$	0						
Central Control	x_1	0	x_3	x_4	x_5	x_6						
Usuarios	Interdependencias						$3x_7$	x_8	0	0	0	0
Camiones							x_7	$2x_8$	x_9	x_{10}	x_{11}	x_{12}
Mantenimiento							0	0	$2x_9$	x_{10}	0	0
Conductores							0	0	0	$4x_{10}$	0	0
Módulos							0	0	x_9	0	$3x_{11}$	0
Ejes Viales							0	0	0	0	0	$4x_{12}$

Fig. 3. Matriz de Interdependencia del Metro y la RTP.

Cuando se crea la matriz de interdependencia del metro y la red de transporte de pasajeros se podrá generar una matriz de 12×12 . En esta matriz se representará cada entidad que interviene en las interdependencias entre el metro y en la red de transporte de pasajeros. En la Figura 4 se tiene la matriz de interdependencia 12×12 . Cada celda de la matriz 12×12 tendrá una ponderación normalizada de acuerdo a la matriz de transición del metro y la red de transporte de pasajeros.

	usuario	Trenes	Instalaciones	Conductores	Mantenimiento	Central Control	Usuarios	Camiones	Mantenimiento	Conductores	Módulos	Ejes Viales
Usuario	1	0.2	0.25	0.08	0	0	x	x	x	x	x	x
Trenes	0.08	0.3	0.17	0.08	0.08	0	x	x	x	x	x	x
Instalaciones	0.4	0	0.67	0	0.08	0	x	x	x	x	x	x
Conductores	0.08	0	0	0.5	0	0	x	x	x	x	x	x
Mantenimiento	0.08	0	0.17	0.08	0.3	0	x	x	x	x	x	x
Central Control	0.08	0	0.08	0.08	0.08	0.08	x	x	x	x	x	x
Usuarios	x	x	x	x	x	x	0.3	0.1	0	0	0	0
Camiones	x	x	x	x	x	x	0.1	0.2	0.1	0.08	0.1	0.08
Mantenimiento	x	x	x	x	x	x	0	0	0.2	0.08	0	0
Conductores	x	x	x	x	x	x	0	0	0	0.33	0	0
Módulos	x	x	x	x	x	x	0	0	0.1	0	0.3	0
Ejes Viales	x	x	x	x	x	x	0	0	0	0	0	0.33

Fig. 4. Matriz Normalizada de Interdependencia 12×12 .

En la Figura 4 se ha normalizado la matriz de interdependencia 12x12, por lo tanto, se toma una celda $x(i, j)$ se le denominará célula central y sus vecinos que son las células $x(i, j - 1)$, $x(i, j + 1)$, $x(i - 1, j)$, $x(i + 1, j)$, $x(i + 1, j - 1)$, $x(i + 1, j + 1)$, $x(i - 1, j - 1)$ y $x(i - 1, j + 1)$.

Para realizar las interdependencias entre el metro y la red de transporte de pasajeros se genera una autómatas celular. La autómatas celular está sujeta a:

$\sum x(\pm i, \pm j) < 3$ no cambia el estado de la célula.

$\sum x(\pm i, \pm j) > 4$ si cambia el estado a 1 de la célula.

Considerando todas las restricciones de las autómatas celulares se analizará lo sucedido en la línea 2 del metro. La línea 2 del metro suspendió su servicio a las 7am con las siguientes restricciones. [9][10][11][12][14][15]

- 44825 usuarios a las 7am.
- 38 trenes en circulación.
- 38 conductores disponibles.
- La red de transporte de pasajeros está sujeta a:
- 731 camiones utilizados de 1458.
- Se propone el 90% de usuarios a las 7am.
- 731 conductores disponibles.

A continuación se propones las siguientes condiciones para el metro y la red de transporte de pasajeros:

- Las instalaciones del metro estén al 75% de su capacidad.
- El personal de mantenimiento del metro estén al 50%.
- Central de Control al 75%.
- El personal de mantenimiento de los camiones estén al 50%.
- 80% de los camiones disponibles en los módulos.
- Los ejes viales estén al 80% de su capacidad.

El diseño de las autómatas celulares correspondientes a la matriz normalizada 12x12 y todas las restricciones se desarrollaron en el software de Matlab versión R2009a. Para realizar dicha simulación se utilizó una PC con las siguientes características:

- Un procesador de 2 núcleos.
- Velocidad del procesador 1.66Ghz
- Memoria Caché 512Kb
- Memoria RAM 2Gb

El lenguaje de Matlab ejecuta comandos o grupos de comandos a la vez sin compilar y genera un código de máquina JIT (del inglés: "Just-In-Time") de la tecnología de compilación, lo que le permite recorrer con rapidez a la solución óptima. [16] En la figura 5 se tiene los estados iniciales de las autómatas celulares.

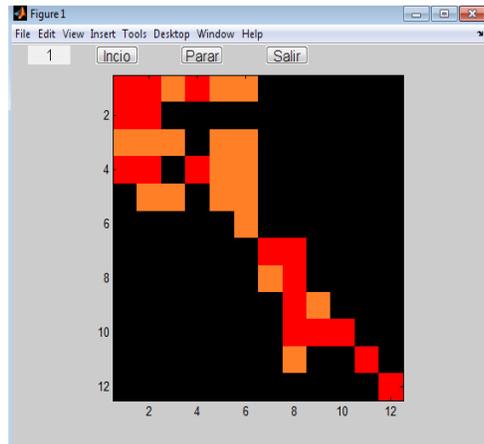


Fig. 5. Estado Inicial de la Autómata Celular.

Con el estado inicial de las autómatas celulares se hacen las iteraciones con base a las restricciones antes mencionadas. En la Figura 6 se observa el estado final de las autómatas celular después de 1100 iteraciones.

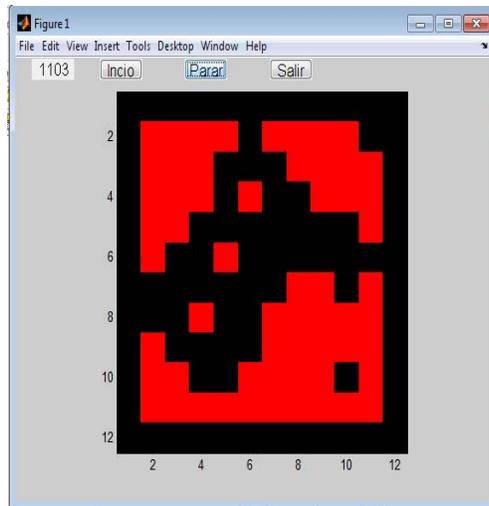


Fig. 6. Estado Final de la Autómata Celular con 1100 iteraciones.

4. Resultados y Conclusiones

Con base en la Figura 6 se analizará las interdependencias del metro y la red de transporte de pasajeros. En la Figura 7 posee una tabulación de la Figura 6 para poder observar las interdependencias.

	usuario	Trenes	Instalaciones	Conductores	Mantenimiento	Central Control	Usuarios	Camiones	Mantenimiento	Conductores	Módulos	Ejes Viales
Usuario												
Trenes												
Instalaciones												
Conductores												
Mantenimiento												
Central Control												
Usuarios												
Camiones												
Mantenimiento												
Conductores												
Módulos												
Ejes Viales												

Fig. 7. Tabulación de las Interdependencias.

Analizando la suspensión del servicio de la línea 2 del metro y con las restricciones que conllevan dicho evento, en la Figura 7 se observa las interdependencias que generaron por el cierre de la línea 2. A continuación se describe las interdependencias entre el metro y la red de transporte de pasajeros.

- Los usuarios del metro afectarán a los usuarios de la red de transporte de pasajeros.
- La cantidad de usuarios del metro demandará más conductores para los camiones de la red de transporte de pasajeros; y a su vez los módulos tendrán que alistar los camiones.
- Los usuarios del metro saldrán a los ejes viales a buscar un transporte para llegar a su destino.
- El mal funcionamiento de los trenes, instalaciones, conductores de trenes, mantenimiento y central de control provocará una aglomeración de usuarios y estos a su vez tendrán que buscar camiones u otro medio de transporte afectando los ejes viales.

- La falta de mantenimiento, una mala sincronización de los camiones, la ausencia o huelga de conductores y/o manifestaciones ciudadanas generará en la red de transporte de pasajeros una aglomeración de usuarios. Por lo tanto, buscarán una alternativa para trasladarse y esto ocasionará una aglomeración con los usuarios del metro y por ende una mayor demanda de trenes para trasladarlos.
- El tráfico, accidentes, construcciones, manifestaciones ciudadanas, etc. en los ejes viales provocará un incremento de usuarios en el metro.

Otro análisis de la suspensión del servicio del metro fue en la línea 1. La suspensión del servicio fue a las 5:50 am [17]. En la Figura 8 y 9 se observa la simulación de las interdependencias del metro y la RTP.

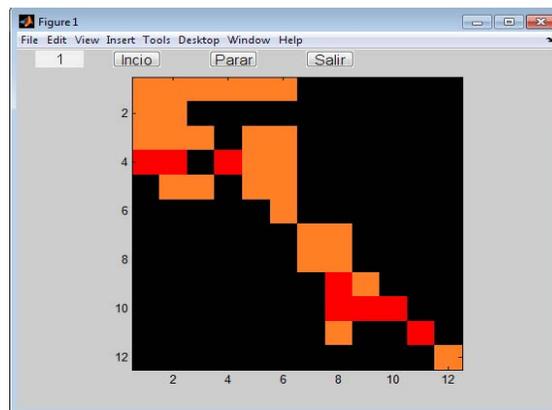


Fig. 8. Estado Inicial de la Autómata Celular.

	Usuarios	Trenes	Instalaciones	Conductores	Mantenimiento	Central de Ctrl.	Usuarios RTP	Camiones	Mantenimiento	Conductores	Módulos	Ejes Viales
Usuarios	█											
Trenes	█	█										
Instalaciones	█	█	█									
Conductores	█	█	█	█								
Mantenimiento					█							
Central de Ctrl.						█						
Usuarios RTP							█					
Camiones							█					
Mantenimiento							█	█				
Conductores							█	█	█			
Módulos							█	█	█	█		
Ejes Viales							█	█	█	█	█	█

Fig. 9. Tabulación de las Interdependencias.

A continuación se describe las interdependencias entre el metro y la red de transporte de pasajeros con base en la Figura 9.

- Los usuarios del metro no afectarán a los usuarios de la red de transporte de pasajeros.
- La cantidad de usuarios no demandará más conductores para los camiones de la red de transporte de pasajeros, por lo tanto, los módulos no tendrán que alistar más camiones que los programados.
- Los usuarios del metro no afectaran a los ejes viales.
- Si continúa el mal funcionamiento de los trenes, instalaciones, conductores de trenes, mantenimiento y central de control provocará una aglomeración de usuarios y estos a su vez tendrán que buscar camiones u otro medio de transporte afectando los ejes viales en la próxima hora.
- La falta de mantenimiento, una mala sincronización de los camiones, la ausencia o huelga de conductores y tráfico o accidentes en los ejes viales generarán conflictos a la central de control, mantenimiento, conductores e instalaciones del metro por no poder transportar a los usuarios del metro.

Con la matriz de interdependencias y las autómatas celular se pueden modelar diferentes escenarios de interdependencias. Esto ayudará a analizar el impacto que tendrá si se colapsa una línea del metro. Las herramientas utilizadas para modelar las interdependencias como i2Sim, solo permiten observar los riesgos de las infraestructuras. Con el arreglo de autómatas celulares proporciona los riesgos y las interdependencias lógicas.

Referencias

1. President's Commission on Critical Infrastructure Protection. 1997, Critica foundations: Protecting America's Infrastructures, 22 febrero de 2011, [http:// www.ciai.gov](http://www.ciai.gov)
2. Decision, P. (2011). Presidential Decision Directive 63. Recuperado el 4 de enero de 2012, de <http://www.ciao.gov>.
3. Protection, P. C. (2001). Report on the Federal Agenda Critical Infrastructure Protection Reserch and Development.
4. A.W. Burks, Von Neumann's self-reproducing automata, in:A.W. Burks (Ed.), Essays on Cellular Automata, University of Illinois Press, Champaign, IL, 1970, pp. 3–64.
5. J. VonNeumann, in:A.W. Burks (Ed.), Theory of Self-Reproducing Automata, University of Illinois Press, Champign, IL, 1966.
6. E.F. Codd, Cellular Automata, Academic Press, New York, 1968.
7. Jarkko Kari, Theory of cellular automata: A survey, Theoretical Computer Science, Volume 334, Issues 1–3, 15 April 2005, Pages 3-33, ISSN 0304-3975

8. Jacques Mazoyer, Ivan Rapaport, Inducing an order on cellular automata by a grouping operation Original Research Article Discrete Applied Mathematics, Volume 91, Issues 1–3, 26 January 1999, Pages 177-196
9. INFOMEX, Solicitud de acceso a la información número: 0322000009212. Distrito Federal.
10. INFOMEX, Gerencia Jurídica número de folio: 0325000051512, Distrito Federal, Información Pública.
11. INFOMEX, Solicitud de acceso a la información número: 0325000049912. Distrito Federal, Información Pública.
12. INFOMEX Respuesta a la solicitud 03200000149 12. Distrito Federal.
13. Rafael Lahoz - Beltrá, Bioinformática Simulación, Vida Artificial e Inteligencia Artificial, Díaz de Santos, Madrid 2004.
14. INFOMEX, Solicitud 010900009612. Distrito Federal. Sub-secretaria de Control de Tránsito.
16. INFOMEX, Solicitud 010900107412. Distrito Federal. Sub-secretaria de Control de Tránsito.
17. Matlab, The language of technical Computing, 11 de Julio del 2012. <http://www.mathworks.com/products/matlab/>
18. Fallas en el Metro. La Razón 2010. Recuperado el 25 de julio de 2013, de <http://www.razon.com.mx:8008/spip.php?article125810>

Identificación de los principios de segmentación y extracción de TRIZ en patrones de diseño

María Erika Auryly García-Cantú, Ulises Juárez-Martínez, Guillermo Cortés-Robles

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de Orizaba
merika.gc@gmail.com, ujuarez@ito-depi.edu.mx,
gc_robles@hotmail.com

Resumen. Los patrones de diseño permiten la encapsulación y reutilización de componentes de software debiendo permitir la sustitución entre patrones. Sin embargo, dicha sustitución muchas veces se dificulta ya que no se tienen principios de diseño definidos para hacerla, muchas veces se hace por la intención del patrón o de acuerdo a la experiencia del diseñador. Por otro lado, TRIZ ofrece, entre otras herramientas, los principios inventivos que permiten la solución de problemas resolviendo contradicciones de forma no convencional. El presente artículo tiene como objetivo presentar las analogías entre los primeros dos principios inventivos de TRIZ (segmentación y extracción) y los patrones de diseño de software.

Palabras clave: Diseño de patrones, Sustitución, TRIZ, Principios inventivos.

1. Introducción

Diseñar software orientado a objetos puede resultar difícil, y aún más diseñar software orientado a objetos que sea reutilizable. Hay que encontrar los objetos pertinentes, factorizarlos en clases, definir interfaces y jerarquías de herencia y establecer relaciones entre clases y objetos [1]. Además, hay que tomar en cuenta los principios de diseño orientados a objetos: Principio de responsabilidad única (SRP), Principio Abierto/Cerrado (OCP), Principio de Sustitución de Liskov (LSP), Principio Inversión de la Dependencia (DIP), Principio de Segregación de Interfaz (ISP), los cuales ayudan a evitar síntomas de mal diseño [2]. Sin embargo, los diseñadores expertos saben que no hay que resolver cada problema desde cero, sino que usan una y otra vez una solución que les funcionó en el pasado. De esta forma se encuentran patrones que resuelven problemas concretos de diseño y hacen que los diseños orientados a objetos sean más flexibles, elegantes y reutilizables. Por lo tanto, un patrón de diseño describe una estructura de diseño que resuelve un problema de diseño particular dentro de un contexto específico [1].

Los patrones de diseño permiten la definición de elementos reutilizables a nivel de diseño y algunos de ellos a nivel de código. Partiendo de aquellos patrones encapsulados, es necesario exhibir interfaces requeridas e interfaces esperadas para su acoplamiento con otros componentes. La sustitución o intercambio de patrones de diseño es factible, sin embargo, muchas veces dicha sustitución está basada en la intención del patrón y no en la interfaz o algún otro principio de diseño, dificultando una verdadera modularidad y evolución de las arquitecturas de software.

TRIZ son las siglas rusas para la Teoría de la Resolución de Problemas Inventivos. El desarrollo de esta metodología fue iniciada en 1946 por Genrikh Altshuller (1926-1998). Se trata de una metodología de resolución de problemas basada en un acercamiento lógico y sistemático desarrollada a partir del análisis de miles de patentes y con el análisis de la evolución de la tecnología. TRIZ se emplea como instrumento intelectual para solucionar problemas técnicos y tecnológicos, sencillos y difíciles, con mayor rapidez y mejores resultados [3].

Para que sea posible resolver un problema con TRIZ, el problema necesita tener por lo menos una contradicción. Si no hay ninguna contradicción técnica, entonces no se trata de un problema inventivo (no es un problema TRIZ) [3].

Una de las herramientas de TRIZ son los principios inventivos, los cuales se utilizan para la solución de contradicciones técnicas y resolver problemas de forma no convencional. El trabajo realizado por Elías Beltrán [4, 5] muestra que es factible establecer mediante los principios inventivos, una forma diferente de aplicar y sustituir patrones de diseño para el software.

Este trabajo presenta las analogías identificadas en los principios inventivos de segmentación y extracción de TRIZ con los 23 patrones de diseño de software, las cuales refinan el trabajo de Beltrán, utilizando principios de objetos, mostrando por qué caen los patrones en cada principio inventivo.

Se presentan en primer lugar los trabajos relacionados, seguidos de la descripción de los principios inventivos y patrones con los que se trabajó para, posteriormente, mostrar los resultados obtenidos. Asimismo, se presenta una sección de discusión donde se explican las diferencias con los trabajos anteriores, además de las conclusiones y trabajo a futuro.

2. Trabajos Relacionados

A continuación se describe el conjunto de trabajos relacionados con el tema.

La aplicación de TRIZ en el ámbito del software es un campo no muy experimentado, hay relativamente pocas publicaciones sobre ello. Se ha aplicado en diferentes áreas como la arquitectura de software [6, 7], en [8] se utiliza como apoyo para el diseño de un software de innovación asistida por computadora. También se toman las soluciones inventivas de TRIZ en algoritmos de procesos de innovación [9]. En [10] presentan 2 casos de TRIZ aplicado a los Servicios de ITs y a la Inteligencia de Ne-

gocios. Asimismo hay un estudio de la teoría de TRIZ para resolver problemas de calidad del software [11]. En [12] se presenta una aplicación de TRIZ y Six Sigma para mejorar el proceso de desarrollo de software. En [13] Nakagawa revisa aspectos de ingeniería de software para ser utilizados con TRIZ; de igual forma, en [14] Kevin Rea estudia la solución de concurrencia mediante TRIZ. Sin embargo, se reportan pocas publicaciones sobre la aplicación de TRIZ en el diseño de software.

En [15] Aarti Goyal et al., describen un método basado en herramientas y principios de TRIZ que aplican para mantener la calidad de los modelos orientados a objetos. Los criterios de calidad fueron mapeados en un conjunto de reglas y prácticas de buen diseño. Se menciona que las herramientas UML no tienen el soporte adecuado para los modelos evolutivos, lo cual resulta en modelos menos mantenibles y reutilizables, y en muchos casos solo se usan como documentación. La evaluación mejorada que proporciona TRIZ será útil en la evaluación de la calidad con anticipación para desarrollar conciencia acerca de los problemas de calidad.

Ma JianHong et al., presentan una investigación relativa al Diseño Orientado a Objetos (Object-Oriented Design, OOD) aplicando TRIZ [16]. Se expresa que la dependencia de las relaciones entre los objetos afecta características importantes de los sistemas como la capacidad de prueba, fiabilidad o facilidad de mantenimiento. Inspirándose en la matriz de contradicciones de la teoría de TRIZ, el documento propone una matriz de contradicción enfocada en el diseño orientado a objetos basándose en el estudio de los patrones de diseño, así como los problemas del diseño de software orientado a objetos. Se construyó una matriz de 22x22 usando 22 parámetros encontrados. Además, se presenta un breve ejemplo de cómo utilizar la matriz para solucionar un problema de diseño. Éste trabajo da la pauta para aplicar TRIZ en el diseño orientado a objetos a partir del análisis de los patrones de diseño, sin embargo, no relaciona los principios inventivos de TRIZ con las características de los patrones, más bien propone una matriz de contradicción parecida a la de TRIZ para solucionar los problemas de OOD.

Una observación temprana de las similitudes entre los patrones de diseño y TRIZ es que ambos sistemas son un conjunto de heurísticas, derivadas de la observación de soluciones exitosas a problemas comunes. Se encuentran fuertes relaciones entre los patrones estructurales de la Gang of Four (GoF) y conjuntos específicos de los principios de TRIZ. En [17] Ellen Domb y John Stamey examinan los siete patrones estructurales, en su nivel conceptual más alto, como instancias de alguno de los 40 principios inventivos de TRIZ y su correspondiente relación con los principios. Aún y cuando sólo se analizaron los patrones estructurales, se hace mención de la intención que se tiene de completar los 23 patrones. Éste documento fue de los primeros en relacionar TRIZ con patrones de diseño, a pesar de que la idea de utilizar TRIZ en el ámbito del software surgió varios años antes.

En [4] Elías Beltrán expone cómo acercar la implementación de patrones de diseño usando principios inventivos de TRIZ, analizando las relaciones que existen entre ambos conceptos abstractos. Pretende generar una herramienta que asista el diseño de software, seleccionando los patrones más convenientes según los principios que se

implementen, lo cual permite mejorar la adaptación y evolución del software. Como hay principios inventivos que se refieren a características físicas y/o químicas, es más difícil encontrar una relación con los patrones de diseño, puesto que el software es lógico. Beltrán explica el análisis que se llevó a cabo para identificar las cinco correspondencias más complicadas entre los conceptos de TRIZ y patrones de diseño, listando los patrones de diseño que se adaptan a la descripción de cada principio inventivo. Posteriormente, el autor presenta una tabla con las relaciones entre los patrones y los principios inventivos, además de un ejemplo de aplicación que muestra cómo utilizar dicha tabla.

De igual forma, Beltrán en [5] propone el uso de los principios inventivos de TRIZ con un enfoque hacia los patrones de diseño de software. El objetivo es probar la capacidad de sustitución entre los patrones de diseño y así fomentar la reutilización de los mismos. Beltrán analiza los 23 patrones de diseño de GoF y los 8 más significativos del paradigma de Programación Orientada a Aspectos (POA). Para lograr el objetivo planteado, primero se analizaron detenidamente cada uno de los principios inventivos de TRIZ así como los patrones de diseño, para después establecer correspondencias con el desarrollo de software. Se logró establecer una relación entre los problemas en el desarrollo de software y los principios inventivos más frecuentes que los solucionan, también se establece una relación entre los principios inventivos y las sugerencias para un buen diseño. Como los patrones de diseño se relacionan entre sí dependiendo de las características que tengan entre ellos, se logró determinar las relaciones existentes. El documento toma dos enfoques: se relacionaron los principios inventivos con los patrones de diseño tomando como base los principios para encontrar qué patrones se relacionaban entre ellos y porqué, pero también se hizo a la inversa, tomando como base los patrones de diseño, para encontrar los principios que se relacionan con cada patrón, así se identificaron los patrones que mantienen en común los mismos principios. De igual forma, se encontraron las similitudes que permiten la sustitución entre patrones usando principios inventivos. El trabajo realizado también proporciona una base para determinar qué patrón de diseño es adecuado para utilizarse de acuerdo a los principios y sugerencias obtenidos con TRIZ.

3. Principios Inventivos y Patrones de Diseño Estudiados

Beltrán [4, 5] realizó un trabajo previo identificando las analogías entre los principios inventivos de TRIZ y los 23 patrones de diseño según GoF [1], sin embargo, dicho trabajo no consideró conceptos como encapsulación, herencia y polimorfismo, y debe ser refinado y documentado para comprobar que el intercambio es factible y así, una vez que se documenten las analogías, desarrollar una guía que facilite y favorezca el intercambio de patrones.

El trabajo de identificar las relaciones entre principios inventivos y patrones requiere un largo análisis, es por esto que se presentan las analogías de sólo 2 principios inventivos.

A continuación se describen los conceptos necesarios para la mejor comprensión del presente trabajo.

Los principios inventivos de TRIZ se usan para resolver contradicciones, se basan en el mismo estudio de patentes y tecnología que desarrolló los patrones de evolución. Por esto, es posible aplicar los mismos principios a problemas de diferentes ámbitos. Existen 40 principios inventivos, sin embargo, sólo se describen los dos primeros, de los cuales se tienen resultados [3].

1. **Segmentación.** Fragmentación. Divide un objeto o sistema en partes independientes. Hace un objeto fácil de desarmar. Este principio tiene 3 casos: A - divide un objeto en partes independientes; B - hace un objeto desmontable para un fácil ensamblado y desensamblado y, C - incrementa el grado de segmentación de un objeto.

2. **Extracción.** Separa la parte (o propiedad) necesaria o elimina una parte que interfiere con el objeto o sistema. El principio presenta 2 casos: A – extraer la parte o propiedad “inquietante” de un objeto; B – extraer sólo la parte o propiedad necesaria de un objeto.

Un patrón de diseño describe una estructura de diseño que resuelve un problema de diseño particular dentro de un contexto específico [1]. A continuación se describen los patrones de diseño de los cuales se identificaron, refinaron y complementaron las analogías hechas por Beltrán [4, 5] con alguno de los dos principios inventivos (los resultados de éstas analogías se muestran en la sección 4).

- *Builder.* Separa la construcción de un objeto complejo de su representación, de forma que el mismo proceso de construcción pueda crear diferentes representaciones.
- *Bridge.* Desacopla una abstracción de su implementación, de modo que ambas puedan variar de forma independiente.
- *Flyweight.* Usa compartimiento para permitir un gran número de objetos de grano fino de forma eficiente.
- *Facade.* Proporciona una interfaz unificada para un conjunto de interfaces de un subsistema. Define una interfaz de alto nivel que hace que el subsistema sea más fácil de usar.
- *Chain of responsibility.* Evita acoplar el emisor de una petición a su receptor, dando a más de un objeto la posibilidad de responder a la petición. Encadena los objetos receptores y pasa la petición a través de la cadena hasta que es procesada por algún objeto.
- *Mediator.* Define un objeto que encapsula cómo interactúan una serie de objetos. Promueve un bajo acoplamiento al evitar que los objetos se refieran unos a otros explícitamente, y permite variar la interacción entre ellos de forma independiente.
- *Proxy.* Proporciona un representante o sustituto de otro objeto para controlar el acceso a éste.
- *Factory Method.* Define una interfaz para crear un objeto, pero deja que sean las subclasses quienes decidan qué clase instanciar. Permite que una clase delegue en sus subclasses la creación de objetos.

- *Abstract Factory*. Proporciona una interfaz para crear familias de objetos relacionados o que dependen entre sí, sin especificar sus clases concretas.
- *Memento*. Representa y externaliza el estado interno de un objeto sin violar la encapsulación, de forma que éste puede volver a dicho estado más tarde.

4. Relación entre Principios Inventivos y Patrones de Diseño

En ésta sección se presentan los resultados de la relación entre los principios inventivos de TRIZ con los patrones de diseño de software obtenidos hasta el momento. Estos resultados logran complementar el trabajo de Beltrán en [4, 5], ya que se realizó un análisis más profundo de cada patrón para conseguir una relación con los principios inventivos 1 y 2, además de quedar documentada la razón por la cual cada patrón presentado tiene relación con alguno de los principios.

Segmentación. El concepto de segmentación se refiere a que sin importar que estén separadas, todas las partes de un conjunto permiten lograr el propósito que se busca. Todas las partes son necesarias. En la Tabla 1 se muestran las relaciones obtenidas entre el principio de segmentación y los patrones de diseño que lograron identificarse.

Tabla 1. Relación del principio de segmentación en patrones de diseño.

Patrón de diseño	Características
<i>Builder</i>	Entra en el principio de segmentación en el caso B porque separa el proceso de construcción. El algoritmo para crear un objeto debe ser independiente de las partes que se compone y cómo se ensambla.
<i>Bridge</i>	Entra en el caso A porque hace la abstracción e implementación independientes y trabajan en conjunto en el nivel más alto de la jerarquía.
<i>Flyweight</i>	Entra en el caso A porque divide los estados de un objeto en intrínseco y extrínseco. Hace las partes independientes, pero a la vez, cada objeto trabaja en conjunto con el objeto intrínseco.

Extracción. El principio de extracción se refiere a la acción de extraer, recuperar o remover un objeto. En la Tabla 2 se muestran las relaciones identificadas entre el principio de extracción y los patrones de diseño.

Tabla 2. Relación del principio de extracción en patrones de diseño.

Patrón de diseño	Características
<i>Facade</i>	Entra en el caso A porque reubica las invocaciones que existen entre las clases que utilizan un subsistema. Las llamadas se pasan de un lugar a otro, a nivel del intercambio de mensajes entre los objetos o clases.
<i>Bridge</i>	Conceptualmente cae en el principio de extracción, estructuralmente en realidad no hay separación, hay una introducción de nuevos elementos para lograr el propósito del patrón.
<i>Mediator</i>	Se encuentra en el caso A, ya que las llamadas se pasan de un lugar a otro, a nivel del intercambio de mensajes. La interacción de objetos es a través del intercambio de mensajes.
<i>Proxy</i>	Se asocia al caso B porque extrae el objeto a sustituir y el proxy controla el acceso a éste. Se considera el concepto de remover porque pasa el objeto de un lugar a otro dando el efecto de posponer el acceso a él.
<i>Chain of responsibility</i>	Cae en el caso B porque extrae la petición (la encapsula) y la mueve entre objetos para que alguno de esos objetos la procese.
<i>Flyweight</i>	Se relaciona con el caso B ya que extrae el estado intrínseco de un objeto, cambiándolo de posición (encapsulación al tenerlo en otro objeto) y lo recupera al momento de compartir ese estado intrínseco con cada uno de los objetos que lo requiera.
<i>Factory method</i>	Se encuentra en el caso B. El patrón maneja una jerarquía paralela lo que permite mover un subconjunto del comportamiento de la jerarquía existente a la nueva. Este comportamiento es la instanciación.
<i>Abstract Factory</i>	Condición: se observa que si en el manejo de la creación de la familia de productos se hace de manera paralela, tiene elementos para considerarse dentro de éste principio. Si no lo hace, no cumple con el principio.
<i>Memento</i>	Presenta extracción como estrategia interna del patrón. Extrae y recupera el estado.

A pesar de que es complicado encontrar una relación entre los principios inventivos de TRIZ con los patrones de diseño, se puede observar que efectivamente existe una correlación entre ambos. Para ello, es necesario comprender correctamente los conceptos de ambas áreas.

5. Discusión

En esta sección se explican las diferencias encontradas en relación con el trabajo de Beltrán [4, 5] y con el de Stamey y Domb [17].

Beltrán reporta *Builder*, *Bridge* y *Visitor* para el principio de Segmentación, sin embargo, de acuerdo al análisis realizado, se descartó el patrón *Visitor* porque no cumple con el objetivo del principio de segmentación; en cambio, se añadió el patrón *Flyweight* debido a que presenta características relativas al propósito del principio. En [17] se reporta el patrón *Bridge* ligado solamente al principio de Extracción, y al *Flyweight* con el principio de transición a una nueva dimensión (principio 17) del cuál aún no se tiene un análisis realizado. No obstante, se demuestra que dichos patrones se pueden asociar con otros principios si se logra comprender a fondo la estructura y propósito del patrón.

De igual forma, Beltrán relaciona los patrones *Facade*, *Flyweight*, *Proxy* y *Chain of Responsibility* con el principio de Extracción. A diferencia de Beltrán, Domb y Stamey relacionan con este principio únicamente al patrón *Bridge*. No obstante, se logró identificar que los patrones *Mediator*, *Factory Method*, *Abstract Factory* y *Memento* también presentan una relación con el principio.

Es importante mencionar que los trabajos anteriores [4, 5, 17] abordan el tema de forma superficial. Sin embargo, el análisis realizado en el presente artículo se hizo desde una perspectiva más a fondo en la parte de Orientación a Objetos, ya que se consideraron conceptos de objetos como encapsulación, herencia y polimorfismo, además de tomar en cuenta la estructura, función y forma en que trabajan los patrones de diseño, lo cual no se considera en los trabajos anteriores. Esto permitió un análisis y comprensión más profundos de los principios, de manera que el trabajo presentado refina y complementa el trabajo previo [4, 5].

6. Conclusiones

Como se comentó a lo largo del presente artículo, los patrones de diseño permiten la reutilización de componentes de software [1]. Por otro lado, TRIZ ofrece los principios inventivos para solucionar contradicciones de forma no convencional [3].

Con base en el trabajo realizado por Beltrán [4, 5] donde se demuestra que los patrones de diseño de software tienen una correlación con los conceptos de los principios inventivos de TRIZ, el presente trabajo toma relevancia ya que encamina al refinamiento en las analogías entre principios inventivos y patrones de diseño al considerar conceptos de objetos que los otros trabajos no contemplan. Esto permite además, obtener un análisis con mayor profundidad de los patrones desde una perspectiva no convencional de diseño y promover su utilización a partir de un enfoque sistemático con los principios inventivos, además de documentar dicho análisis. Asimismo, presenta un alto grado de innovación puesto que aún y cuando TRIZ se usa en diferentes áreas de ingeniería [14], en el ámbito de los sistemas de software no se reportan trabajos realizados con la profundidad ni el alcance que tiene el presente trabajo.

7. Trabajo a Futuro

Analizar y documentar la correlación de los 38 principios inventivos de TRIZ restantes (principios 3 al 40) con los patrones de diseño de software.

Determinar qué principios son congruentes con las tendencias de evolución estáticas, dinámicas y/o cinemáticas con el objetivo de plantear tendencias de evolución en la sustitución de patrones.

Agradecimientos

Se agradece al Consejo Nacional de Ciencia y Tecnología (CONACyT) y a la Dirección General de Educación Superior Tecnológica (DGEST) por el apoyo brindado para la realización del presente trabajo.

Referencias

1. Gamma, E., Helm, R., Johnson, R., Vlissides, J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.
2. R. C. Martin, M. C. Feathers, T. R. Ottinger, J. J. Langr, B. L. S. J. W. Grenning, and K. D. Wampler, Clean Code: A Handbook of Agile Software Craftsmanship. Robert C. Martin Series, Upper Saddle River, NJ: Prentice Hall, August 2008.
3. K. Rantanen and E. Domb, Simplified TRIZ. , Auerbach Publication: Auerbach Publication, 2008.
4. Elías Beltrán, Ulises Juárez, Guillermo Cortés. Patrones de diseño de software con principios inventivos. 7o Congreso Iberoamericano de Innovación Tecnológica & Desarrollo de Productos 2012, Orizaba, Veracruz. Noviembre 2012.
5. Elías Beltrán, Ulises Juárez, Guillermo Cortés. TRIZ en el desarrollo de arquitecturas de software. 6° Congreso Iberoamericano de Innovación Tecnológica 2011, Querétaro, Querétaro. Octubre 2011.
6. Daniel Kluender. TRIZ for software architecture. ScienceDirect, Procedia Engineering, Elsevier Ltd. vol. 9, pp. 708-713, 2011.
7. Song-Kyoo Kim. Design of Enhanced Software Protection Architecture by Using Theory of Inventive Problem Solving. In Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management, 2009. IEEM 2009. IEEE Computer Society, Hong Kong, 8-11 Dec. 2009, pp.978-982.
8. Cao Guozhong, Guo Haixia, Yu Jiang, Tan Runhua. Computer Aided Product Innovation Software Based on Extended-Effect and its Application. In Proceedings of the Software Engineering, 2009. WRI World Congress WCSE '09, vol.2, IEEE Computer Society, Xiamen, China, 19-21 May 2009, pp.63-67.

9. Brad, M. Fulea, B. Mocan, A. Duca, E. Brad. Software Platform for Supporting Open Innovation. In Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics, AQTR 2008, vol.3, IEEE Computer Society, Cluj-Napoca, Romania, 22-25 May 2008, pp.224-229.
10. Kas Kasravi. Applications of TRIZ to IT: Cases and Lessons Learned. In Proceedings of the TRIZCON 2010. Altshuller Institute, Dayton, Ohio. Oct. 2010.
11. Su-Hua Wang, Durgesh Samadhiya, Dengjie Chen. Software Development and Quality Problems and Solutions by TRIZ. In Proceedings of the International Symposium on Frontiers in Ambient and Mobile Systems (FAMS-2011). Vol. 5. Procedia Computer Science, Ontario, Canada, 2011, 730-735.
12. Garikapati Pavan Kumar. Software Process Improvement –TRIZ and Six Sigma (Using Contradiction Matrix and 40 Principles). TRIZ Journal. CTQ Media. April 2005.
13. Toru Nakagawa. Software engineering and TRIZ – Structured Programming Reviewed with TRIZ. In Proceedings of TRIZCON 2005, Altshuller Institute, April 2005.
14. Kevin C. Rea. Using TRIZ in Computer Science – Concurrency. TRIZ Journal, CTQ Media. August 1999.
15. S. B. Goyal, Aarti Goyal, Pratima Sharma, Neha Singhal. Analyzing Object Models with Theory of Innovative Solution. In Proceedings of the 2012 Second International Conference on Advanced Computing & Communication Technologies (ACCT '12). IEEE Computer Society, Washington, DC, USA, 46-50.
16. Ma JianHong, Zhang Quan, Wang Yanling, Zhang Wei. Research and Application of the TRIZ Contradiction Matrix in OOD. In Proceedings of the WRI World Congress on Software Engineering, 2009. WCSE '09, vol.3, IEEE Computer Society, Xiamen, China, 19-21 May 2009, pp.247-251.
17. John W. Stamey, Ellen Domb. Workshop: communicating design patterns with TRIZ. In Proceedings of the 24th annual ACM international conference on Design of communication (SIGDOC '06). ACM, New York, NY, USA, 2006, 131-133.

Bon iterative graph feature mining for graph indexing

A. Pankaj Moses Monickaraj¹, K. Vivekanandan², D. Ramya Chitra³

¹Doctoral Scholar, Department of Computer Science, Bharathiar University, India

²Professor, BSMED, Bharathiar University, India

³Assistant Professor, Department of Computer Science, Bharathiar University, India

pankajmoses@hotmail.com

Abstract. Extracting Sub graph from a colossal graph database is one of the key problems in Graph Mining. Feature mining approaches (i.e) mine at once algorithms like G-index, FG index, gSpan build index for any graph data. But, if incase of changes in the data (updates reach or change in memory size), the graphs have to be upgraded as well. Quiet obviously the index has to be updated. It is quite expensive to construct and deploy a new Graph Index from scratch instead iterative mining algorithms can be used. As the sizes of these algorithms are small, they can be iteratively used then and while at consistent intervals. These provide the updated list of features from the graph database. Iterative graph feature mining algorithm [1] is one of the key algorithms playing dominant role in feature extraction. After extracting the updated features [3], they have to be re-indexed and inserted in proper into the index for which various search algorithms can be used. In this paper, an improved iterative graph feature mining is proposed and tested with AIDS and e-molecules datasets which is further compared with the existing results.

Keywords: Graph, Graph Mining, Graph Indexing.

1. Introduction

In the recent days scientific and technological advances have revealed that plenty of certain data or uncertain data or may be patterns can be modeled as graphs. As a result, it is of special interest to process graph containment queries effectively on large graph databases. Consider a graph database G , let q be a query (i.e) the set sub-graph relevant to the query q . All the subgraphs relevant to the query q is retrieved in G which contain q as subgraph(s). The isomorphism testing can be done in case if the number of subgraph are vast and any of the indexing mechanism [1] – [5] can be opted to index the complete data in G so that it can be easily accessible. In [6], a framework is pictured with indexing, iterative mining [7] and re-indexing. In fig [2] in [7], an iterative algorithm was used to extract the updated graph features. In this paper, an improved iterative mining algorithm is proposed and tested with two different datasets. (i.e) AIDS antiviral dataset and e-molecules dataset.

This paper is organized as follows: we introduce the background and preliminaries in Section II. Then we present the set of related work in section III. In section IV, we first introduce the complete overview of Iterative algorithm. Then in section V, we propose Bon Iterative algorithm to show the improved search space. The algorithm is tested with AIDS and e-molecule datasets in Section VI and the results are discussed. Finally, we conclude this paper in Section VII.

1.1 Preliminaries

Two graphs say $g_1 = (v_1, e_1, l_1)$ and $g = (v_2, e_2, l_2)$ are isomorphic to each other if there is a bijection which is a mapping such that a pair of adjacent vertices u_1, v_1 in g_1 is mapped to a pair of adjacent vertices u_2, v_2 in g_2 where $L(u_1) = L(u_2)$, $L(v_1) = L(v_2)$ and $L(E(u_1, v_1)) = L(E(u_2, v_2))$ and vice versa.

Let G be a graph database, a subgraph g_1 is a frequent subgraph iff it maintain the level greater than the defined parameter level, minimum level $\lfloor |D| \rfloor$. The number of sub graphs which are obtained for a query from G would be the $\lfloor |D| \rfloor$ for the query. A subgraph search is calculated by adding filtering and verifying paradigm.

Let $f = \{f_1, f_2, \dots, f_m\}$ be the set of graph features. Graphs from G can be vectorized and are represented as an m dimensional vector $Xg = [x_1, x_2, \dots, x_m]$, where $x_i = 1$ if $f_i \subset g$ and $x_i = 0$. Each vector will be the core features which will be used building index.

The candidate set can be obtained [7] from

$$C_q = \text{find}(p_i) = \{f_i \in \text{maxsubgraph}(q, f) \mid \text{nd}(p_i) = D(\text{maxSubgraph}(q, f))\} \quad (1)$$

$$\text{Maxsub}(q, f) = \left\{ f_i \in \frac{f}{f_i} \subset q, \nexists x \in g \text{ s. t. } p_i \subset x \subset q \right\} \quad (2)$$

(Maximum Sub graph for a query q)

For set queries say Q ,

$$\text{minSup}(f, Q) = \{q \in Q \mid f \in \text{maxSub}(q, f)\} \quad (3)$$

The time taken for processing for a query q ,

$$T_{\text{resq}} \rightarrow (q) = T_{\text{filter}} \rightarrow (q) + T_{\text{verif}}(C(q)) \quad (4)$$

The experiment is implemented and tested with AIDS and e-molecule datasets in Section 5.

2. Related work

Depth first search code tree is used by $gSpan$ [8] which extracts the features through their subgraph from a graph database. $gSpan$ also prefers on from the initial right most vertices and proceeds further. Xifeng Yan et.al has proposed $gindex$ [10]

which uses the basic sub structure or sub graph whose parameters are greater than the minimal fixed parameter say $|D|$. Almost all the features present in the graph database G are taken under initial consideration and further they are extended as the graph with related loops through $gIndex$ [10]. GraphSig [11] initially converts the data into feature vectors and further proceeds on mining the sub feature vectors. Iterative feature mining [7], finds a subgraph feature p with maximum domination the maximum and minimum support can be flexibly altered according to the features perspective present in the concern data. In this paper, the performance of the iterative algorithm is tested with the proposed algorithm and tested with datasets.

3. Bon iterative algorithm for indexing

Consider a graph database G , a stable index is obtained by using any of the mines at once algorithms. Consider there is an up gradation in data. In such case, the instead of completely scanning the dataset just the up gradations can be obtained through Bon Iterative Algorithm.

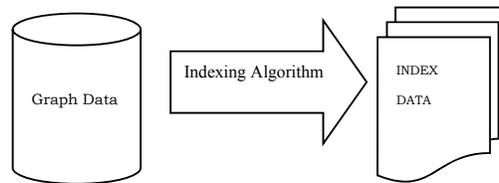


Figure 1. Graph Indexing.

Consider a graph index with feature set p_{n-1} of size $n-1$, find a new graph feature p , where p does not belong to p_{n-1} such that expectation of the verification cost be T_{verif} . This new features is $\{p, p_{n-1}\}$ is indexed until it is generated with $C(q)$, where $C(q)$ is the candidate set of the query q . Finally after adding the new feature the set becomes into p_n .

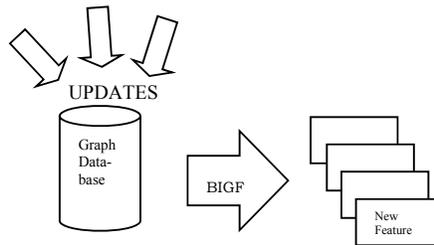


Figure 2. New feature extraction from updated database.

The new feature should be selected as the one with maximizing support through Dijkstras algorithm such that the feature attains the minimal support so as it can be added into the index.

Minimal Support:

Consider a set of query Q and a subgraph feature $p_0 \in P$, a graph $q_0 \in Q$ is a minimal super query of the feature p_0 iff the query q_0 has a maximal feature of p_0 .

$\text{minSup}(p_0, Q) = \{q_0 \in Q | p_0 \in \text{maxSub}(q_0, P)\}$. by the minimal super queries of p_0 .

This is how the graph features are extracted to move into index. BIGFMA is tested further with datasets and their experimental results continue in the upcoming section.

3. Experimental results

3.1 Datasets aids antiviral screen dataset

To test the scalability, database index construction etc, we test with the dataset from National Cancer Institute AIDS antiviral, 3 classes (B) July 29, 2004 By Fei Yuan. The potency data is at http://ntp.nci.nih.gov/docs/aids/aids_data.html. NCI A and NCI B have slightly different sets of molecules and different descriptor sets. Otherwise, the potencies are the same.

3.2 E-molecule dataset

To test the scalability, database index construction etc., we have also tested with dataset from chemical page consisting of molecules structure (<http://www.e-molecules.com/>)

3.3 Screen shots and graphs aids dataset

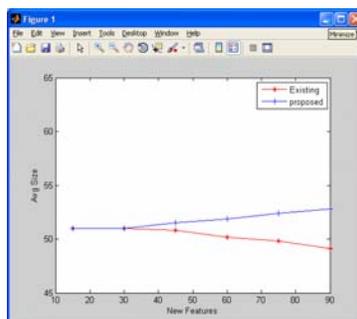


Figure 3. Candidate set size between IGFMA vs BIGFMA.

Bon iterative graph feature mining for graph indexing

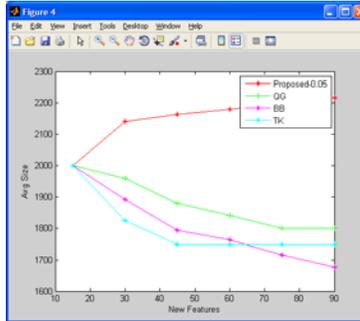


Figure 3. Bon iterative mining at DF(0.05).

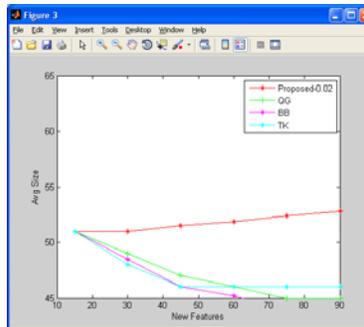


Figure 4. Bon iterative mining at DF(0.02).

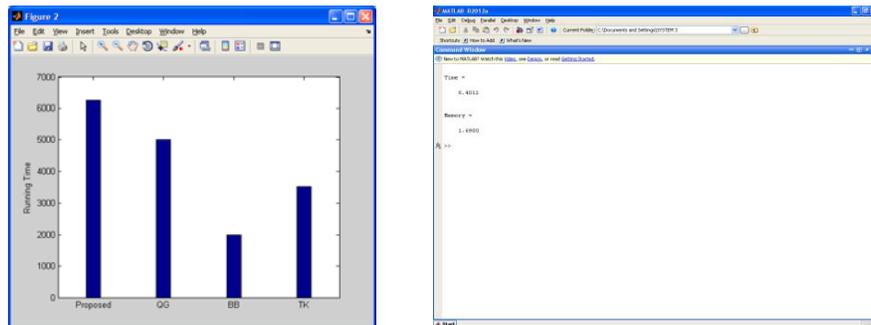


Figure 5. Running time and memory usage of Bon iterative algorithm.

Table 1. Feature count with decreasing MIN-support.

	.1-.8	.8-.6	.6-.4
Proposed	46	103	221
Existing	54	104	247

The counts of the features that are mined are compared with the existing and the proposed algorithm. In case of (.1-.8) 14.8 % of improvement, there is for .8-.6 little improvement and for 6-.4 10.53% of improvement. Hence for this dataset Bon Iterative Algorithm performs better than the existing Iterative algorithm. The time taken for the execution of the algorithm was 6.4012 seconds.

3.4 Molecule dataset

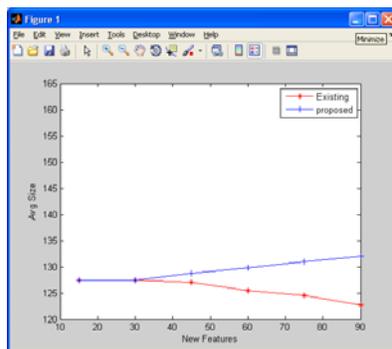


Figure 6. Candidate set Size between IGFMA vs BIGFMA.

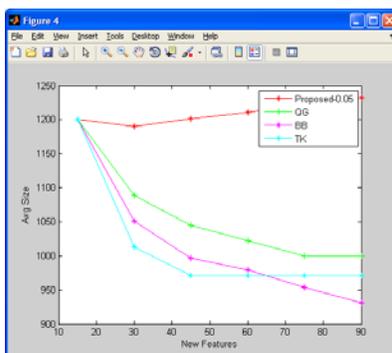


Figure 7. Bon Iterative Mining at DF(0.05).

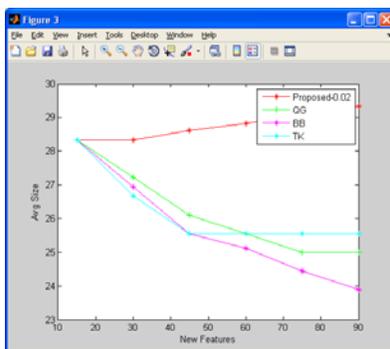


Figure 8. Bon Iterative Mining at DF(0.02).

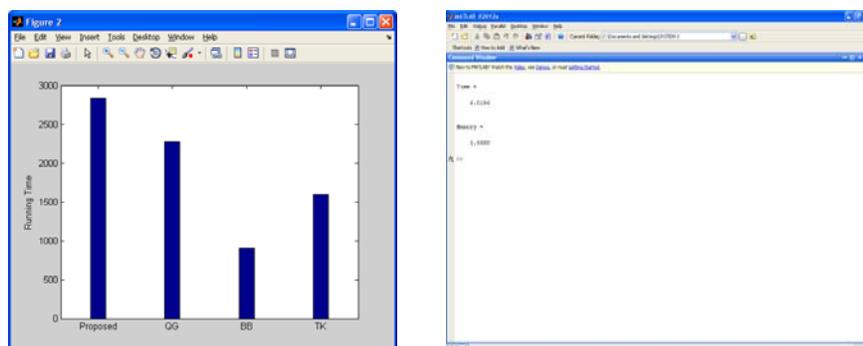


Figure 9. Running time and memory usage of Bon Iterative Algorithm.

Table 2. Feature count with decreasing MIN-support.

	.1-.8	.8-.6	-.4
Proposed	48	98	220
Existing	54	104	247

In case of e- molecules dataset, the minimal support are found between proposed and existing 11.12 %, 5.7 % and 10.93 % were the improvements found. The time taken for the execution of molecule dataset was 6.0196 seconds.

Environmental requirement: Front end designed with MATLAB (r2012a) with i3 processor and 80 GB memory. RAM size is 4 GB with Intel mother board.

4. Conclusions

Here we have investigated the working of Iterative mining of graph features for the subgraph search problem. From table 1 and from Figure 3,4,5,6 the Proposed Bon Iterative Graph feature Mining Algorithm is providing a better result than the existing Iterative Algorithm when tested with the AIDS dataset. Also, when tested with E-molecules dataset, from table and figure: 7, 8, 9, 10 they provide a better result in time complexity and memory complexity. Hence BIGFMA is better than the Iterative graph feature Mining algorithm for feature extraction and indexing.

Acknowledgement

We would also like to thank Mr. M. Pradeep who helped in programming throughout this work. Would like to thank our friends Mr. Ramkumar, Doctoral Scholar, Ms. Preethi Rajavel, Department of Microbial Bio-Technology, Bharathiar University, Coimbatore- 46, Tamil Nadu, India for their valuable tips during implementation.

References

1. B. Sun, P. Mitra, and C. L. Giles. Irredundant informative subgraph mining for graph search on the web. CIKM, 2009.
2. J. Cheng, Y. Ke, W. Ng, and A. Lu. Fg-index: Towards verification-free query processing on graph databases. SIGMOD, 2007.
3. S. Zhang, M. Hu, and J. Yang. Treepi: A novel graph indexing method. ICDE, pp.966–975,2007.
4. P. Zhao, J. X. Yu, and P. S. Yu. Graph indexing: tree + delta \leq graph. VLDB, 2007, pp. 938– 949.
5. X. Yan, P. S. Yu, and J. Han. Graph indexing: a frequent structure-based approach. SIGMOD, 2004.
6. K. Vivekanandan, A. Pankaj Moses Monickaraj, D. Ramya Chithra. Graph Mining Sub Domains and a Framework for Indexing – A Graphical Approach. International Journal of Advanced Computer Science and Applications, Vol. 4, No.4, 2013.
7. Dayu Yuan, Prasenjit Mitra, Huiwen Yu, C. Lee Giles. Iterative Graph Feature Mining for Graph Indexing. IEEE 28th International Conference on Data Engineering, 2012.
8. Xifeng Yan Jiawei Han, gSpan. Graph-Based Substructure Pattern Mining, ICDM 2003. Proceedings, 2002.
9. X. Van, P. S. Yu, and J. Han. Graph indexing: a frequent structure-based approach. Proc. of the ACM SIGMOD international conference on Management of data, pages 335-346, 2004.
10. S. Nijssen and J. N. Kok. The gaston tool for frequent subgraph mining. Electronic Notes in Theoretical Computer Science, Vol. 127, No. 1, pp. 77 – 87, 2005.
11. S. Ranu and A. K. Singh. Graphsig: A scalable approach to mining significant subgraphs in large graph databases. ICDE '09, 2009, pp. 844–855.

Desarrollo de una aplicación de Software para simulación de maquinado en un torno básico CNC para el Instituto Tecnológico de Orizaba

Héctor Uriel Osorio Dávila, Hilarión Muñoz Contreras

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de Orizaba
huod.1210@gmail.com, hmunozc189@msn.com

Resumen. La simulación es una rama de la computación, la cual conlleva en la inclusión de varias disciplinas dependiendo del sistema a representar mediante un ambiente controlado. A través de los años y con la mejora tanto en los componentes de hardware y software es posible enriquecer cualquier sistema de información, siendo esto una ventaja para realizar los sistemas de simulación lo más parecidos al entorno real posible. Este trabajo propone el desarrollo de un sistema de simulación de maquinado mediante torno básico CNC, con el objetivo de fortalecer los conocimientos prácticos de los alumnos del Instituto Tecnológico de Orizaba en las carreras de Ingeniería Industrial e Ingeniería Mecánica, formando así profesionistas mejor preparados para el mundo laboral en la manufactura.

Palabras clave: Simulación, CNC, Torno.

1. Introducción

Actualmente el uso de la tecnología informática en cualquier rama es imprescindible, debido a que la implementación de sistemas de información ayuda a la automatización de procesos industriales, farmacéuticos, contables, médicos entre otros. Dentro del ámbito académico, el uso de herramientas tecnológicas para impartir una preparación más profesional es completamente necesaria; la necesidad de formar profesionistas competentes para el mundo laboral es una obligación de cualquier institución educativa. Las herramientas tecnológicas necesarias dentro de una institución educativa varían en cada especialidad que oferte la institución. Dentro del Instituto Tecnológico de Orizaba se cuenta con las carreras de Ingeniería Industrial e Ingeniería Mecánica, las cuales tienen las materias Máquinas- Herramientas y Maquinado Asistido por Computadora, estas materias demandan una preparación practica para la manufactura industrial la cual consiste en el diseño y elaboración de piezas utilizando Máquinas-Herramientas tales como : torno, fresa, taladro, cepillo entre otros como se describe en [1]; las carreras antes mencionadas se topan con el inconveniente no contar

con los recursos materiales necesarios para llevar a cabo prácticas reales que aporten experiencia a los alumnos. La falta de recursos idóneos para realizar prácticas de manufactura industrial no solo es un problema que tiene esta institución educativa, muchas instituciones atraviesan por la misma situación. Una de las formas para dar solución al problema antes mencionado es utilizar software de simulación para maquinado industrial, dicho maquinado es un proceso de manufactura en la cual se usa una herramienta de corte para remover el exceso de material de tal manera que el material remanente sea la forma de la parte deseada[1]. Los sistemas de simulación de maquinado generalmente lo utilizan las grandes empresas para la capacitación de su personal y evitar así desperdicios de material, una posible avería en las máquinas o un accidente con el personal que manipule las máquinas-herramientas; todas estas actividades al final se reflejan en pérdidas económicas para la empresa, es por ello que invierten gran recurso económico para obtener sistemas de simulación de maquinado debido a las ventajas que ofrecen como las descritas en [2], pero ¿qué sucede con las instituciones educativas?. Generalmente las instituciones educativas no cuentan con el presupuesto necesario para adquirir las licencias correspondientes a los sistemas de simulación para maquinado, quedando delimitados a utilizar versiones de prueba muy limitadas, es por ello que este trabajo propone el desarrollo de una aplicación de software para simulación de maquinado en un torno básico CNC, el cual es un torno controlado por una computadora que ejecuta programas controlados por medio de datos numéricos [3].

2. Trabajos Relacionados

El área de la simulación por computadora tiene un campo de aplicación bastante extenso, para el desarrollo de este proyecto de investigación se tomaron en cuenta los trabajos más significativos relacionados a la simulación de manufactura industrial. En [4] se describe la importancia de sistemas CNC (Control numérico por computadora) en la manufactura industrial para fabricar piezas, las máquinas que utilizan un sistema CNC tienen mayor ventaja sobre las máquinas manuales, puesto que los cortes, traslados o perforaciones son más precisas bajo el mando de una computadora que sobre el manejo manual de estas. El campo de aplicación para la manufactura de piezas y herramientas es muy amplio, es por eso que el autor propone el desarrollo de un sistema de simulación 3D para máquinas CNC. En [5] se presenta un modelo para determinar el movimiento de una máquina de 5 ejes y se analizan los modelos más precisos de la herramienta basada en cinemática con el objetivo de aplicarlos en cualquier máquina CNC, debido al conflicto que produce el que cada máquina utilice sus propios movimientos, por tanto este trabajo propone realizar un modelo más genérico aplicable a cualquier máquina de 3 o más ejes. Por otra parte en [6] se describen las características de las nuevas generaciones de máquinas CNC, las cuales son: portabilidad, interoperabilidad y adaptabilidad; para alcanzar estas características se propone el nuevo estándar STEP-NC, el cual reemplazará a los códigos G&M, ya que amplía las capacidades geométricas, la tolerancia de datos, eliminara los cuellos de botella ocasionados por los códigos G&M. Otro de los trabajos sobresalientes es [7], en este trabajo se realiza un estudio de simulación para máquinas CNC que utiliza la realidad

aumentada para complementar el mundo real con la información virtual, la inclusión de realidad aumentada enriquece la experiencia del usuario y favorece una mejor adaptación a entornos reales. Actualmente los sistemas de simulación son usados para la capacitación de personal en muchos ámbitos. Todos los sistemas virtuales tienen un grado de eficacia aceptable y resuelven distintas problemáticas, entre las más comunes se encuentra la exposición del personal a entornos peligrosos, la pérdida económica por manejo incorrecto en alguna máquina manufacturera, por mencionar algunos. En [8] se realiza un análisis de tareas cognitivas que afectan en el desarrollo de un sistema de simulación CNC, entre las cuales se incluye la motivación de los expertos en el uso de la máquina CNC para lograr mayor eficacia en los sistemas virtuales de simulación, contemplando situaciones cotidianas en un entorno de trabajo. En [9] se presentan varios métodos para utilizar VRML (Virtual Reality Modeling Language) usándola como la tecnología para integrar sistemas de fabricación virtual. VRML es un lenguaje de modelado para la realidad virtual en la web, lo cual potencializa su alcance, dentro y fuera de donde sea implementado. El uso de tecnologías actuales como la computación gráfica 3D hacen posible la realización de sistemas de realidad virtual dedicados a la capacitación. En [10] se presenta un sistema de entrenamiento virtual para personal operario de máquinas CNC desarrollado con World Tool Kit (WTK), este sistema surge debido a la necesidad de capacitar correctamente y eficientemente al personal operario de máquinas CNC y conseguir así una mejor manipulación en dichas máquinas. Otro de los trabajos más relevantes es [11], aquí se presenta una plataforma de software llamada WorkCellSimulator que permite gestionar un entorno de simulación. El objetivo de este programa es ayudar al usuario a definir los procesos de producción potenciales para automatizar. El [12] propone el desarrollo de herramientas de simulación para máquinas CNC mediante el estándar STEP-NC, el cual es un nuevo modelo de datos de alto nivel que hace los entornos de simulación más completos, desechando así todas las limitantes que se encuentran en los estándares de bajo nivel G&M CODE. En [13] se presenta un panorama de la demanda actual por la fabricación rápida y eficiente de herramientas, para lo cual se realiza un estudio acerca de las deficiencias de los estándares actuales como G-CODE y de cómo el estándar STEP-NC lo soluciona. Propone un marco de trabajo que se basa en el descubrimiento de conocimiento usando minería de datos para derivar reglas y controlar el proceso de fabricación.

3. Arquitectura del Sistema CNC

El primer sistema de control numérico computarizado (CNC) fue desarrollado desde la década de 1950, el objetivo era y sigue siendo reemplazar la manipulación manual de máquinas como: tornos, cepillos, fresas entre otros y utilizar una serie de instrucciones codificadas escritas en códigos G&M para la manipulación computarizada de las máquinas descritas anteriormente [14]. Las ventajas que ofrece el sistema CNC son:

- La planificación, flexibilidad y programación

- Estimación de costo
- Precisión
- Eficiencia
- Productividad
- Seguridad

Los elementos que conforman un sistema CNC se describen en la figura 1:



Figura 1. Arquitectura básica CNC

La manera cómo interactúan los elementos mostrados en la figura 1 es la siguiente:

- La computadora local es la encargada de gestionar el código G&M, lo cual significa el ingreso instrucciones que representen el mecanizado de la pieza deseable.
- La unidad de control es el corazón del sistema CNC, debido a que en esta parte la información descargada de la máquina local es manipulada usando la lógica de hardware y software para finalmente almacenarla en memoria con las instrucciones para dirigir las operaciones de maquinado de acuerdo a los códigos G&M.
- Finalmente la máquina CNC es la encargada de realizar el mecanizado físico de las piezas, dirigido por la unidad de control.

4. Sistema de Simulación

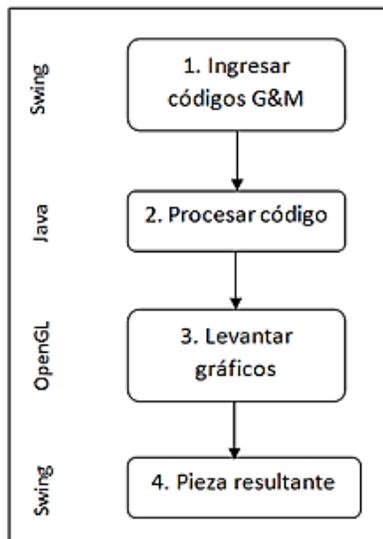
Para el desarrollo de este sistema de simulación se contemplaron diferentes tecnologías, tanto de código abierto como privado; se eligió el uso de herramientas de código abierto debido a las prestaciones que estas nos aportan. Como lenguaje de programación se tiene Java y como API para gráficos 3D se eligió OpenGL. Java es un lenguaje de programación orientado a objetos desarrollado por SunMicrosystem [15], recientemente adquirido por Oracle; es un lenguaje independiente del sistema operativo gracias a su máquina virtual muy potente y versátil. Por otra parte OpenGL (Open Graphics Library) es considerado como la principal API para la realización de gráficos en 3D, fue desarrollada por Silicon Graphics Inc. en 1992 [16]; es una API multilenguaje y multiplataforma para desarrollar aplicaciones en 2D y 3D, contiene más de 250 funciones diferentes para pintar escenas tridimensionales muy complejas a partir de primitivas geométricas simples como líneas, puntos y triángulos, además de que es muy utilizado en áreas como CAD (diseño asistido por computador), realidad virtual, simulación entre otros. Para el desarrollo de este sistema de simulación además de dominar las herramientas tecnológicas (lenguaje de programación y api) para programar la aplicación es fundamental comprender y dominar el funcionamien-

to de las máquinas CNC, especialmente de los G&M CODE (códigos g y m) que son el lenguaje estándar que utilizan las máquinas CNC para realizar operaciones como : corte, perforación, traslado, velocidad, aplicación de refrigerante entre otros. Con el dominio de estos códigos será posible realizar "rutas de trazo" que interpretaremos con el lenguaje java y representaremos con OpenGL.

El flujo de trabajo planteado para la realización del sistema de simulación consiste en 4 etapas:

1. Se ingresarán los valores correspondientes a los códigos G&M mediante un formulario en la interfaz de usuario.
2. Se realizará el procesamiento de los códigos ingresados en la etapa uno, éste procesamiento utilizará la lógica con la cual una máquina CNC interpreta los códigos G&M.
3. Se realizará la representación gráfica de los códigos procesados en la etapa dos.
4. Se mostrará la figura resultante en tres dimensiones sobre una interfaz gráfica.

El flujo de trabajo descrito anteriormente se representa en la figura 2.



Figur 2. Flujo de trabajo.

Para el desarrollo del sistema de simulación se han propuesto distintas interfaces gráficas para el usuario, teniendo en cuenta aspectos como la facilidad de uso, que la aplicación sea intuitiva, pero sobre todo que sea funcional; como prototipo para la interfaz gráfica de usuario se tiene la figura 3 que muestra la interfaz propuesta hasta éste momento:

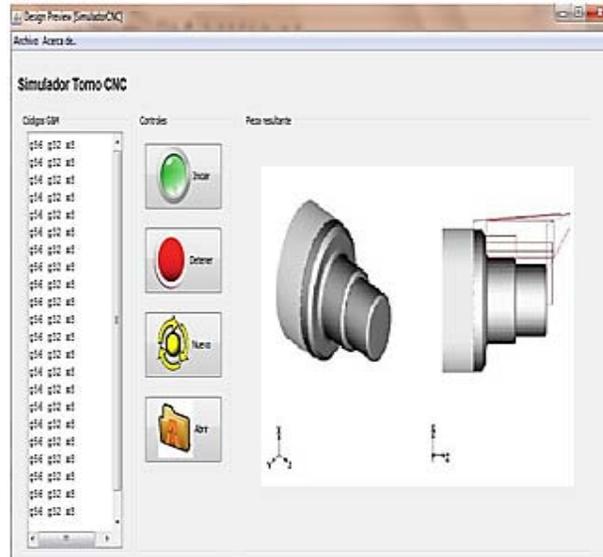


Figura 3. Interfaz gráfica de usuario.

5. Trabajo a Futuro

Actualmente uno de los retos más importantes en el área de la simulación de maquinado es realizar sistemas funcionales y de fácil manipulación; consiguiendo así el interés de quienes lo utilicen y que la experiencia en dicho simulador sea placentera y benéfica al instruir correctamente en la manipulación de máquinas-herramientas de un entorno real. Es por ello que para el desarrollo de este sistema de simulación se tiene contemplado la inclusión de VRLM que como se describe en [9] proporciona grandes beneficios para la realización de entornos virtuales y con ello explotar las capacidades para realizar un entorno industrial lo más parecido a la realidad posible. Por otra parte también se tiene contemplado analizar el desarrollo del simulador de maquinado utilizando tecnologías web, debido a que no se descartan las ventajas que ofrece tener una aplicación web contra una aplicación cliente/servidor.

6. Conclusiones

La simulación por computadora es una rama de la informática que ha ganado importancia en diversas áreas gracias a las ventajas que ofrece, como lo es la capacitación de personal entre muchas otras. Para el ámbito industrial específicamente manufactura en máquinas-herramientas es de gran importancia, pues permite lograr un alto

nivel de manipulación para la operación correcta de dichas máquinas y así evitar posibles averías por causa de un mal uso, o desperdiciar material lo cual se traduce en pérdidas para cualquier empresa que implemente máquinas-herramientas. Las grandes industrias manufactureras tienen los medios para adquirir software de simulación, pero las instituciones educativas difícilmente tienen dichos productos de software cómo es el caso de la institución donde se realiza este trabajo. Es por ello que este trabajo tiene gran importancia, pues con la herramienta ya terminada que propone este proyecto, esa limitante para la institución queda descartada al tener su propio software simulador. El principal beneficio que aporta se da en el lado académico, puesto que los alumnos obtienen una preparación con mayor calidad y por ende adquieren mejores bases para el mundo laboral.

Agradecimientos

Este artículo fue realizado gracias al patrocinio del Consejo Nacional de Ciencia y Tecnología (CONACYT) y a la Dirección General de Educación Superior Tecnológica (DGEST).

Referencias

1. Mikell P. Groover, Fundamentos de Manufactura Moderna: Materiales, Procesos y Sistemas, Pearson Education, 1997.
2. Simulación computarizada: Un camino hacia el incremento de la productividad en los sistemas de manufactura, 03 - 2013, <http://www.galeon.com/henrymontano/articulo.htm>.
3. Tooling U-SME, What is the definition of torno de CNC?, 03 - 2013, <http://www.toolingu.com/definition-301110-32206-torno-de-cnc.html>
4. Zongmin Chen, Development of OpenGL Based 3D Simulator for Computer Numerical Control. In Proceedings of the 2010 International Conference on Artificial Intelligence and Computational Intelligence - Vol. 3. 2010. IEEE Computer Society, Washington, DC, USA, pp. 319-321.
5. Stephen Mann, Sanjeev Bedi, Gilad Israeli, Xiaoran (Linda) Zhou. Machine models and tool motions for simulating five-axis machining. *Comput. Aided Des.* 42, 3 (2010), pp.231-237.
6. X. W. Xu and S. T. Newman. Making CNC machine tools more open, interoperable and intelligent-a review of the technologies. *Comput. Ind.* 57, 2 (2006), pp.141-152.
7. J. Zhang, S. K. Ong, A. Y. C. Nee. A multi-regional computation scheme in an AR-assisted in situ CNC simulation environment. *Comput. Aided Des.* 42, 12 (2010), Pp.1167-1177.
8. Dimitris Nathanael, George-Christopher Vosniakos, Stergios Mosialos. Cognitive task analysis for virtual reality training: the case of CNC tool offsetting. In Proceedings of the 28th Annual European Conference on Cognitive Ergonomics (ECCE '10). ACM. 2010, New York, NY, USA, pp.241-244.

9. Tsai Sung ,Ching Ou. Using Virtual Reality technologies for manufacturing applications. *Int. J. Comput. Appl. Technol.* 17, 4 (2003), pp.213-219.
10. Wang Xiaoling, Zheng Peng, Wei Zhifang, Sun Yan, Luo Bin, Li Yangchun. Development an interactive VR training for CNC machining. In *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry (VRCAI '04)*. ACM, New York, NY, USA, pp.131-133.
11. Stefano Tonello, Guido Piero Zanetti, Matteo Finotto, Roberto Bortoletto, Elisa Tosello, Emanuele Menegatti. WorkCellSimulator: a 3d simulator for intelligent manufacturing. In *Proceedings of the Third international conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAN'12)*, Itsuki Noda, Noriaki Ando, Davide Brugali, James J. Kuffner (Eds.). 2012. Springer-Verlag, Berlin, Heidelberg, pp.311-322.
12. Yu Zhang, Xiao-Lan Bai, Xun Xu, Yong-Xian Liu. STEP-NC based high-level machining simulations integrated with CAD/CAPP/CAM". *Int. J. Autom. Comput.* 9, 5 (2012), Pp.506-517.
13. Sanjeev Kumar, Aydin Nassehi, Stephen T. Newman, Richard D. Allen, Manoj K. Tiwari. Process control in CNC manufacturing for discrete components: A STEP-NC compliant framework. *Robot. Comput.-Integr. Manuf.* 23, 6 (2007), pp.667-676.
14. Shanshan Chen, Graphical Simulation Software For The Tool Path Within An Integrated Cad/Cam/Cnc Environment, Faculty of Texas Tech University,1995.
15. Java Team Development, The Java Language Specification, 04 - 2013 ,<http://docs.oracle.com/javase/specs/jls/se7/html/index.html>.
16. OpenGL Oficial, 04 - 2013, http://www.opengl.org/wiki/Getting_Started.

Aplicaciones colaborativas con realidad aumentada para dispositivos móviles basados en Android

Joaquín Llanillo Barquet, María Antonieta Abud Figueroa,
Ignacio López Martínez, Celia Romero Torres

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de Orizaba
joaquinllanillo@gmail.com,
{mabud, ilopez, cromero}@ito-depi.edu.mx

Resumen. La realidad aumentada es una tecnología de reciente impacto que consiste en un sistema conformado por cámaras digitales y sensores que superponen contenido digital al mundo físico en tiempo real. Desarrollar aplicaciones de Realidad Aumentada colaborativa es una actividad en aumento, sin embargo no existen arquitecturas de software definidas para este tipo de proyectos. Las aplicaciones colaborativas de realidad aumentada con enfoque educativo son verdaderas herramientas en la interacción Profesor-Alumno ya que permiten compartir un mismo entorno físico y manipular objetos 3D superpuestos al mundo real. En este artículo se propone el desarrollo de una arquitectura base para el desarrollo de aplicaciones colaborativas con realidad aumentada basadas en el Sistema Operativo Android.

Palabras clave: Arquitectura de software, Android, Aplicaciones colaborativas.

1. Introducción

La realidad aumentada consiste en una combinación de tecnologías que permiten mezclar contenido generado por computadora con un entorno visual del mundo real. La realidad aumentada se distingue de la realidad virtual debido a que la realidad virtual emplea de forma completa un entorno y objetos 3D mientras que en la realidad aumentada el entorno es tomado de la realidad y los objetos virtuales son el aporte de la realidad aumentada. La realidad aumentada en la actualidad es una tecnología que tiene aplicación en muchas áreas como lo son el entretenimiento, la arquitectura, la medicina, la educación, por mencionar algunas, incrementando su uso debido a la posibilidad de utilizarlas en dispositivos móviles. *Android* es un Sistema Operativo de uso libre con una plataforma móvil totalmente personalizable; se basa en *Linux* y se encuentra respaldado por una amplia comunidad de desarrolladores dedicados a la creación de aplicaciones que se ejecutan bajo esta plataforma. Por otra parte, las aplicaciones colaborativas tienen importancia en distintas áreas de la industria y la inves-

tigación; muchas empresas reconocen la importancia del trabajo en equipo y día a día buscan facilitar la comunicación entre sus colaboradores por medio de tecnología y aplicaciones de software. La información y el conocimiento son componentes esenciales para establecer relaciones de comunicación. Las aplicaciones colaborativas buscan actividades que se realizan en grupos o equipos de personas y permiten integrar el conocimiento con la tecnología de forma compartida para los usuarios. Estas aplicaciones, a menudo denominadas “*groupware*” emplean técnicas que facilitan el trabajo en grupo debido a que mejoran el rendimiento y tienen la capacidad de realizar actividades a distancia y al mismo tiempo por medio de redes de comunicación. Los roles en las aplicaciones colaborativas no solo representan de forma estática colecciones de usuarios, sino también el dinamismo que ejecuta dichas aplicaciones.

Durante el tiempo de ejecución de una aplicación, los roles reaccionan de forma flexible a la dinámica inherente de forma colaborativa. En la actualidad las aplicaciones colaborativas toman importancia debido a que permiten la interacción de varios usuarios en una misma aplicación en tiempo real, esto en conjunto con tecnologías de realidad aumentada comprende una aplicación interactiva no solo entre varios usuarios sino también entre los mundos real y virtual que intervienen en esta tecnología. Las arquitecturas de software son abstracciones de los componentes que realizan tareas específicas, las interfaces y la comunicación que existe entre ellos, éstas se implementan en dispositivos con arquitecturas de hardware específicas. Las arquitecturas de software son el resultado de ensamblar los componentes de la aplicación de forma que se logre el objetivo de desarrollar un producto de software que cumpla con todos los requisitos funcionales y no funcionales. La definición de una arquitectura para el desarrollo de estas aplicaciones toma importancia en la necesidad de estandarizar el desarrollo de este tipo de aplicaciones para lograr una eficiente interacción entre diferentes dispositivos móviles con características de hardware diferentes. Este documento presenta una propuesta arquitectónica para el desarrollo de aplicaciones colaborativas con Realidad Aumentada, incluyendo el análisis de las tecnologías propuestas.

2. Trabajos Relacionados

Diversos trabajos se han propuesto para el desarrollo de aplicaciones colaborativas con Realidad Aumentada. A continuación se describen los más relevantes.

Construct3D [1] es una aplicación enfocada en la enseñanza de matemáticas y geometría en los niveles medio superior y superior. Brinda un conjunto básico de funciones para la construcción de gráficos primitivos así como diversas operaciones. *Construct3D* emplea la tecnología de Realidad Aumentada. Este proyecto se encuentra basado en la tecnología “*Studierstube*” empleando un dispositivo *HMD* (*Head Mounted Display*) capaz de superponer imágenes generadas por computadora tomadas del mundo real.

En [2] se presenta un prototipo de realidad aumentada colaborativa para la capacitación en la industria con las siguientes características: a) Información virtual como la anotación de objetos virtuales. b) Espacio de trabajo personalizado. c) Permite a usuarios remotos, compartir el espacio de trabajo de realidad aumentada de un usuario de una computadora portátil, ver lo que él está viendo y observar sus acciones como si los usuarios remotos se encontraran presentes físicamente. d) los usuarios remotos manipulan e interactúan con los objetos virtuales que forman parte del espacio de trabajo de la realidad aumentada como si se tratara de objetos físicos reales. e) Los usuarios realizan conversaciones de audio en tiempo real. Para el manejo de realidad aumentada se emplean marcadores.

En [3], se presenta el sistema *MAGIC*, el cual apoya el trabajo llevado a cabo por los arqueólogos. Los arqueólogos constantemente cambian de lugar, además colaboran con otras personas por lo que *MAGIC* se convierte en un sistema de Realidad Aumentada Colaborativa. Se emplea un *Gateway* (puerta de enlace) entre los dos ambientes. La información del entorno real se transfiere al mundo virtual por medio de la cámara que lleva el usuario. El arqueólogo visualiza los objetos descubiertos y sus especificaciones si éstos se encuentran disponibles y almacenados en la base de datos. El recorrido aumentado es particularmente útil para ver los objetos y conocer a que categoría pertenecen. En [4] se evalúan los diferentes usos que se le dan a la comunicación en un teléfono móvil.

La comunicación se ve como una interacción multidimensional reforzada mediante la actividad colectiva. El contexto investigado en este trabajo es la comunicación y la interacción social en una sesión de un juego multijugador para dispositivos móviles. El juego descrito fue un ensayo multi-usuario de código abierto denominado "*First Strike*" desarrollado por *Nokia*. El entorno de juego se compone de un lugar físico de juego, los jugadores con sus teléfonos móviles y una pantalla pública.

En [5] se evalúan cuatro juegos para dispositivos móviles que emplean Realidad Aumentada. Es apropiado considerar la combinación del aprendizaje por medio de juegos de realidad aumentada ya que aprovechan y toman el contexto del mundo real. Con el fin de obtener un mejor contexto de los jugadores, el dispositivo móvil de cada jugador está dotado de un conjunto de sensores tales como *GPS* (Sistema Global de Posicionamiento), *RFID* (Identificador de Frecuencias de Radio), entre otros tales como *Bluetooth*, Infrarrojos y cámara.

En [6] se analiza *FAR-Play*, una plataforma de apoyo para el desarrollo de *AARGs* (*Augmented/Alternate Reality Games*). Requiere instalar dos aplicaciones móviles, el navegador de realidad aumentada *Layar* proporciona el contenido de realidad aumentada para los *AARGs* y el lector de código *QR BeeTagg* para ampliar los juegos a los espacios interiores. El estado del juego se mantiene en un repositorio a través del cual, las *APIs*, el cliente móvil, el mundo virtual y el sitio Web tienen acceso a la información del juego. La comunicación entre las aplicaciones y el motor de juego, se lleva a cabo a través de llamadas mediante *HTTP* a las *API* de *REST*. *Layar* permite dejar la lógica de juego bajo el control del servidor. Para la localización del jugador dentro de un ambiente determinado se emplea *Bee Tagg7*, un lector de código QR. La comuni-

cación entre sistemas se realiza a través del intercambio de archivos *JSON* (*JavaScript Object Notation*) tanto para comunicaciones internas como externas.

En [7] se describe el proyecto “*Who do you think you really are?*” (¿Quién crees que eres en realidad?) Del Museo de Historia Natural de Londres; consiste en un sistema para uso de cámaras de estudio de seguimiento y el uso de marcadores por medio de *ARToolKit* para la incorporación de objetos virtuales. proporciona experiencia multisensorial interesante y útil a nivel pedagógico.

En [8] se describe una arquitectura de software basada en componentes que acelera la tarea de desarrollo y despliegue de aplicaciones de colaboración en los dispositivos móviles; esta arquitectura otorga un papel importante a la conectividad entre dispositivos y en la Realidad Aumentada colaborativa. Este proyecto demuestra que es esencial para los desarrolladores concentrarse en aspectos de diseño de alto nivel. El proyecto incluye una biblioteca para proporcionar acceso independiente de la plataforma a los recursos del sistema y simplificar el desarrollo orientado a objetos.

En [9] se describe el proyecto *Studierstube* que implementa una arquitectura de software basada en el patrón arquitectónico *Cliente-Servidor* el cual mantiene una base de datos donde se gestionan los objetos virtuales y los datos de visualización. Los usuarios se conectan al entorno del servidor a través de una red por medio del software del *cliente*; éste obtiene una copia de los datos del servidor, los cuales son empleados para procesar la imagen que se mostrará al usuario. Para el caso de uso colaborativo o concurrente de la aplicación, se mantiene la copia de los escenarios (color de los objetos, posición, entre otros.) de forma consistente y si ocurre un cambio en dichos escenarios, deberán reflejarse en las copias de cada uno de los usuarios.

3. Arquitectura

En esta sección se presenta la definición de la arquitectura, la cual se basa en los patrones arquitectónicos Cliente-Servidor y Modelo-Vista-Controlador.

3.1 Planteamiento de la arquitectura

En la figura 1 se muestran las capas de la arquitectura propuesta. El aspecto colaborativo de este tipo de aplicaciones se presenta en los componentes encerrados en un recuadro con línea punteada. A continuación, se describe cada una de las capas que conforman la arquitectura propuesta.

La capa de presentación del Cliente se compone de la siguiente forma:

- a) Capturar la imagen: a través de la cámara del dispositivo se adquiere la imagen a procesar.
- b) Presentar la Realidad Aumentada: consiste en mostrar la imagen real con el objeto virtual superpuesto.

La capa de presentación del Servidor se compone de la siguiente forma:

- a) Habilitación de usuarios: consiste en registrar a los usuarios que van a interactuar con la aplicación.

La capa lógica del Cliente está conformada por cinco componentes:

- a) Reconocimiento del marcador: Se busca la información del marcador en el archivo binario específico.
- b) Reconocimiento de la acción del usuario: Se identifica las acciones que el usuario en turno realiza con el marcador.
- c) Obtención del elemento a aumentar: Se busca información del objeto virtual que selecciona el usuario.
- d) Mezclado: Es el encargado de lograr que el mundo real converja con el objeto virtual
- e) Comunicación: Este componente se ocupa de interactuar con el servidor proporcionándole información de las acciones del participante.

La capa lógica del Servidor está conformada por cinco componentes:

- a) Actualizar vista de los participantes: Se encarga de identificar cual de los usuarios que colaboran en la ejecución de la aplicación realiza las actividades en turno.
- b) Comunicación: Este componente se ocupa de interactuar con el cliente indicándole la gestión de los turnos, información de los demás participantes, y las posibilidades del juego en tiempo de ejecución.

La capa de datos del Cliente cuenta con dos componentes:

- a) Catálogo de marcadores: Es un repositorio con los archivos binarios que contienen la información de los marcadores.
- b) Catálogo de objetos virtuales: Es un repositorio con los objetos virtuales que se aumentarán cuando el usuario seleccione el marcador.

La capa de datos del Servidor cuenta con un componente:

- a) Catálogo de usuarios registrados: Consiste en un repositorio con los datos de los usuarios activos de la aplicación.

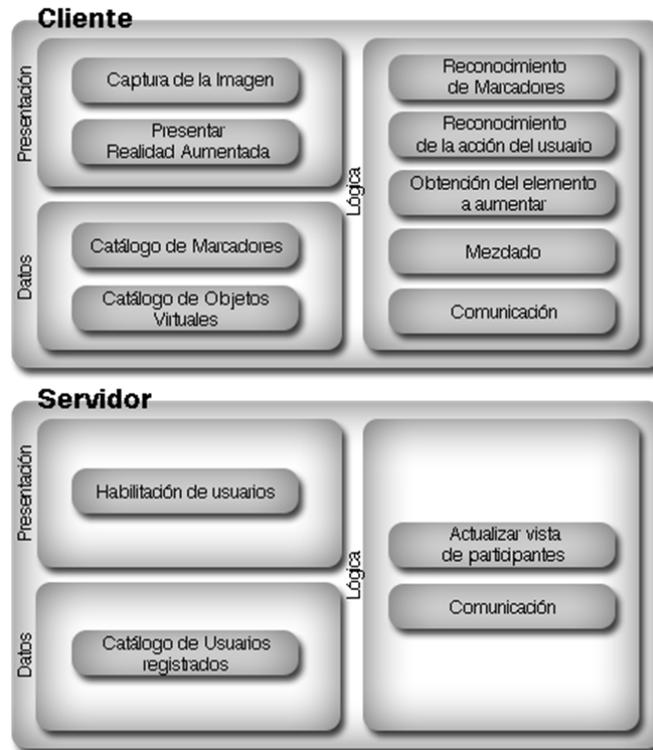


Figura 1. Prototipo de la propuesta de arquitectura.

3.2 Tecnologías para la implementación

Las aplicaciones de realidad aumentada para dispositivos móviles están relacionadas con un sinnúmero de tecnologías que abarcan tanto Sistemas Operativos como bibliotecas de realidad aumentada. A continuación se describen algunas de las tecnologías analizadas que se relacionan con este trabajo.

Android

Este Sistema Operativo[10] para dispositivos móviles es de uso libre con una plataforma totalmente personalizable, incluye *middleware* y aplicaciones esenciales; el *Kit* de desarrollo de software (*SDK*) de *Android* proporciona las herramientas y las interfaces de programación (*API's*) necesarias para comenzar a desarrollar aplicaciones en la plataforma con el lenguaje de programación *Java*.

iOS

Este sistema operativo tiene una licencia propietaria con un *SDK* para desarrolladores registrados. Funciona con un núcleo basado en *Unix*. El lenguaje de programación empleado es *Objective-c* siendo éste una extensión del lenguaje *C* que permite trabajar en un modelo orientado a objetos. Cuenta con un entorno de desarrollo llamado *XCode* el cual permite la creación de la interfaz de usuario por medio de *storyboards* y el uso de herramientas para medir el desempeño de las aplicaciones desarrolladas, el entorno incluye un simulador virtual entre otras características [10].

Vuforia

Vuforia es una extensión de Realidad Aumentada creada por *Qualcomm* incluida en el IDE *Unity3D*. Es un sistema de marcadores basado en realidad aumentada que permite la detección y el seguimiento de la visión. El IDE *Unity3D* permite a los desarrolladores crear aplicaciones de Realidad Aumentada y Juegos. *Vuforia* se integra por los siguientes componentes: una cámara fotográfica, la imagen del convertidor, *tracker* y el video de fondo *render*, la aplicación de código y los recursos de destino [11].

ARToolkit

ARToolkit es una biblioteca de código abierto que permite la creación de aplicaciones de realidad aumentada. Esta biblioteca, emplea las capacidades del seguimiento de video calculando en tiempo real la posición de la cámara y la orientación de los marcadores físicos que indican la ubicación de los objetos que se superponen al mundo real. *ARToolkit* soluciona los problemas de la realidad aumentada que son: 1. El seguimiento de punto de vista. 2. Interacción entre objetos virtuales [12].

FLARToolkit

Es una biblioteca para Flash de código abierto para el desarrollo de aplicaciones de Realidad Aumentada. *FLARToolkit* detecta un marcador con la cámara Web y calcula su posición en un espacio tridimensional. La Realidad Aumentada requiere bibliotecas *ActionScript de Flash*, *Papervision3D* y el uso de *Adobe Flex Builder* para el desarrollo de la aplicación de Realidad Aumentada. Con *FLARToolkit*, el usuario mantiene el marcador en el enfoque de la cámara Web y con ello lograr la animación de los objetos en pantalla. *FLARToolkit* combinado con *Papervision3D* logra que los usuarios vean objetos virtuales aumentados en el mundo real [13].

En la Figura 2, se presenta el esquema de colaboración de las aplicaciones, se observa cómo el acceso a la aplicación se realiza a través de diferentes tipos de dispositivos.

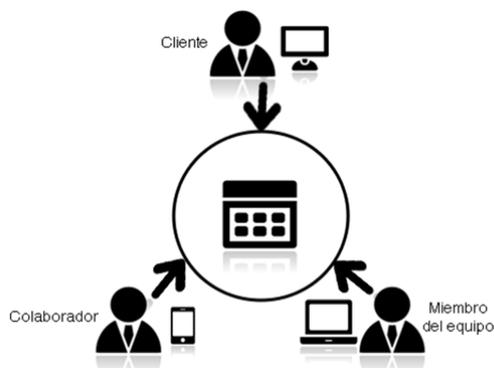


Figura 2. Representación de colaboración en las aplicaciones.

3.3 Justificación de la solución

Para la implementación de la arquitectura, se propone utilizar el Sistema operativo Android y la biblioteca de realidad aumentada Vuforia, esto debido a que en la actualidad existe un sinnúmero de dispositivos basados en Android a diferencia de iOS donde solo es posible desarrollar aplicaciones para dispositivos móviles de Apple. Para el caso de la biblioteca de Realidad Aumentada de Vuforia, se eligió como alternativa debido a que ésta no es tan estricta respecto a la definición de los marcadores independientemente que es la más actual que se encontró. Para comprobar el funcionamiento de la arquitectura propuesta se pretende elaborar un prototipo que simule el juego de Memorama. Las cartas estarán representadas por marcadores, teniendo cada una de ellas asociado el objeto 3D a aumentar. Cuando uno de los jugadores señale alguna de las tarjetas, la aplicación desplegará su objeto 3D correspondiente, el cual se visualizará en los dispositivos de todos los participantes en la sesión de juego. La imagen desaparece a los diez segundos después de que deje de ser señalada. Cuando los objetos de ambas cartas coinciden, la aplicación deshabilita los marcadores de dichos objetos para todos los jugadores. En la figura 3, se muestra una representación de lo que se pretende lograr con este proyecto.

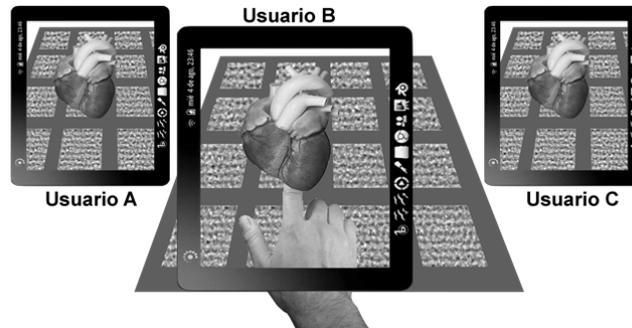


Figura 3. Representación gráfica de una aplicación colaborativa.

4. Trabajo a Futuro

Como trabajo a futuro se pretende implementar los componentes de la arquitectura en una primera versión con el Sistema Operativo Android y la biblioteca de realidad aumentada Vuforia y se pretende desarrollar el prototipo del juego memorama descrito en la sección anterior. Se realizarán pruebas con diversos dispositivos móviles. Evaluar la arquitectura con aplicaciones propias del Sistema Operativo de iOS y desarrollar aplicaciones educativas y evaluar su efectividad en escenarios reales.

5. Conclusiones

En la actualidad existen aplicaciones Colaborativas de Realidad Aumentada basadas en el Sistema Operativo Android, sin embargo existe poca información sobre arquitecturas de software como base para el desarrollo de las mismas, lo que convierte este proceso de desarrollo en una actividad tediosa en la interacción de los dispositivos. Conforme a lo analizado en este documento, se concluye que el diseño de una arquitectura de software para aplicaciones de realidad aumentada colaborativa con enfoque educativo permite a los desarrolladores de estas aplicaciones contar con una base que logre de forma efectiva la interacción entre diferentes dispositivos móviles basados en Android.

Agradecimientos

Este trabajo fue patrocinado por el Consejo Nacional de Ciencia y Tecnología de México.

Referencias

1. Hannes Kaufmann, Collaborative Augmented Reality in Education, Imagina Conference, 2003, Monaco Mediaw, Monaco, pp. 0-4.
2. Xiao Wei Zhong, Pierre Boulanger, Nicolas D. Georganas, Collaborative Augmented Reality: A Prototype for Industrial Training, 21th Biennial Symposium on Communication, 2002, IEEE Computer Society, Canada, pp. 0-5.
3. Philippe Renevier, Laurence Nigay, Mobile Collaborative Augmented Reality: the Augmented Stroll, 8th IFIP International Conference, 2001, Springer Berlin Heidelberg, Toronto, Canadá, pp. 299-316.
4. JaanaLeikas, Hanna Strömberg, VeikkoIkonen, RikuSuomela, JuhaniHeinilä, Multi-User Mobile Applications and a Public Display: Novel Ways for Social Interaction, Fourth Annual IEEE International Conference on Pervasive Computing and Communications, 2006, IEEE Computer Society, Pisa, Italia, pp. 70-75.
5. F. Fotouhi-Ghazvini, R. A. Earnshaw, D. Robison, P. S. Excell, Designing Augmented Reality Games for Mobile Learning using an Instructional-Motivational Paradigm, CyberWorlds International Conference, 2009, IEEE Computer Society, Bradford, pp. 312-319.
6. L. Gutierrez, I. Nikolaidis, E. Stroulia, S. Gouglas, G. Rockwell, P. Boechler, M. Carbonaro, S. King, fAR-PLAY: a framework to develop Augmented/Alternate Reality Games Pervasive Computing and Communications Workshops, 2011, IEEE Computer Society, Seattle, pp. 531-536.
7. Ailsa Barry, Jonathan Trout, Augmented Reality in a public Space: The Natural History Museum, London, 2012, Vol. 45, IEEE Computer Society, London, United Kingdom, pp. 42-4.
8. Daniel Wagnet, Thomas Pintaric, Florian Ledermann, Dieter Schmalstieg Towards Massively Multi-User Augmented Reality on Handheld Devices, 2005, Vol. 3468, Springer Berlin Heidelberg, Munich, Germany, pp 208-219.
9. Szalavári, Z., Schmalstieg, D., Fuhrmann, A., Gervautz, M. Studierstube: An environment for collaboration in augmented reality, 1998, Vol. 3, Springer-Verlag, pp. 37-48.
10. Mark H. Goadrich, Michael P. Rogers, Smart smartphone development: iOS versus android, 42nd ACM Technical Symposium on Computer science education, 2011, ACM, New York, USA, pp. 607-612.
11. Guiran Chang, Chunguang Tan, Guanhua Li, Chuan Zhu, Developing mobile applications on the Android Platform, 2010, Springer-Verlag, Berlin, Heidelberg pp. 264-286.
12. Kato, Hirokazu, Virtual Object manipulation on a table-top AR environment, Augmented Reality (ISAR 2000) Symposium, 2000, Hiroshima Univ., Japón, pp. 111-119.
13. Padmavathi S, Medicherla, George Chang, Patricia Morreale, Visualization for increased understanding and learning using augmented reality, International conference on Multimedia Information retrieval, 2010, ACM, New York, United States, pp. 441-444.

Aplicación de un navegador 3D con una interfaz de usuario natural utilizando la plataforma .NET, SDK y Kinect de Microsoft

I. Peredo Valderrama¹, K. Anaya Rivera¹, R. Peredo Valderrama²

¹Universidad Politécnica de Querétaro, Carretera Estatal 420 S/N, El Rosario el Marqués Querétaro CP. 76240

²Escuela Superior de Cómputo, Instituto Politécnico Nacional, Unidad Profesional Adolfo López Mateos, Av. Juan de Dios Bátiz S/N Esq. Miguel Othón de Mendizábal, Col. Linda Vista, Ciudad de México, D.F., C.P. 07738, México.
{ivan.peredo;karina.anaya}@upq.mx; rperedo@ipn.mx

Resumen. Kinect está a la vanguardia de los sistemas conocidos como Interfaz de Usuario Natural (NUI), que permiten una interacción más natural entre un usuario y la computadora a través de los movimientos del cuerpo y la voz. Esta es la próxima generación de la interacción hombre-máquina (MMI), desarrollando nuevas aplicaciones para interactuar con las personas y la computadora de una manera más natural. Las aplicaciones que utilizan estas nuevas NUIs están relacionadas con temáticas tales como: negocios, educación, juegos de video, y aún más allá, amplían los horizontes de desarrollo de aplicaciones. En este trabajo se ofrece una propuesta de bajo costo para diseñar, desarrollar e implementar un navegador 3D con una NUI usando Kinect SDK, NET y el motor de Unity (Unity Engine). Permitiendo aumentar la interacción hombre-máquina, esta propuesta pretende ofrecer una herramienta de apoyo con una innovadora NUI para el aprendizaje a través de prácticas virtuales reduciendo los costes y riesgos inherentes. Un navegador 3D con una NUI ofrece una simple y fácil manera de manipular objetos virtuales.

Palabras clave: Interface natural de usuario, Microsoft Kinect, navegador 3D, Plataforma SDK.

1. Introducción

En la actualidad estamos viviendo cambios muy rápidos con la tecnología en casi todos campos, el desarrollo de las Interfaces de Usuario para los diferentes tipos de aplicaciones no ha sido la excepción, lo que ha provocado su desarrollo a lo largo de los años. Steve Mann desarrollo a lo largo de los 70's y hasta los 90's diferentes tipos de Interfaces de Usuario alternativas a las tradicionales: línea de comando y por medio del ratón a través de una Interfaz de Usuario Grafica (Graphical User Interface, GUI por sus siglas en inglés), este nuevo tipo de interfaces las denomino Interfaz de Usuario Natural (Natural User Interface, NUI por sus siglas en inglés). Las NUI son la siguiente generación de Interfaces de Usuario [1], las cuales nos permiten interactuar con los usuarios de formas impensables hace algunos años, usando diferentes tipos de entradas como: rastreo de movimiento, voz, etc., las NUI nos posibilitan nue-

vas maneras de interactuar con los diversos tipos de computadoras como: teléfono inteligente, PC, laptop, iPad, etc., a través de una interfaz natural que explote las habilidades que hemos adquirido a lo largo de nuestra vida.

Una de los éxitos más sonados de Nintendo en los últimos años ha sido la consola Wii [2], causando un boom en el segmento de los videojuegos, vendiendo hasta el momento la consola Wii cerca de 100 millones de unidades [3]. Una de las causas de su éxito se debió a su innovadora NUI manipulada a través de un sensor de movimiento denominado Wii remote, Nintendo desarrollo una NUI que permitía interactuar con la consola en un espacio tridimensional real de una manera radicalmente novedosa. Ante el enorme éxito de Nintendo y su consola Wii, la competencia implemento NUIs similares a las de Nintendo, la consola Playstation 3 desarrollo el sensor de movimiento denominado Move [4], y la consola Xbox 360 desarrollo el sensor de movimiento libre denominando Kinect [5]. Esto ha renovado el interés de desarrollo de NUIs avanzadas e interactivas para los usuarios para los diferentes tipos de aplicaciones.

El éxito del Kinect colocando más de 10 millones de unidades vendidas en sus primeras semanas de venta, lo convirtieron en el accesorio de mayor venta en la historia de las consolas de videojuegos, conformando una amplia base instalada, posibilitando que fuera visto por muchos desarrolladores como una nueva herramienta para desarrollar innovadoras NUIs debido a su bajo costo y a la liberación del Kit de Desarrollo de Software (Software Development Kit, SDK por sus siglas en inglés) [6], posibilitándonos crear NUIs innovadoras que respondan a los movimientos y a la voz humana, posibilitándonos comunicarnos de una manera más natural con las computadoras, de una manera muy similar a como nos comunicamos las personas. Debido a los motivos anteriormente mencionados y al gran soporte que nos facilita la empresa Microsoft para el Kinect [6-7], fue que se eligió el dispositivo para el desarrollo de la presente propuesta. También se utilizó en la presente propuesta el motor de renderizado de la empresa Unity Technologies [8], debido al conjunto integrado de herramientas intuitivas de desarrollo, rápidos flujos de trabajo para desarrollar contenido interactivo 3D, su facilidad de publicación multiplataforma, sus activos (assets) de calidad y su activa comunidad de desarrollo y compartición de conocimiento. Estas dos tecnologías son utilizadas ampliamente en la presente propuesta como herramientas de desarrollo bajo la plataforma .NET.

En el presente trabajo se presentan las diferentes etapas que se requirieron para el correcto desarrollo del Navegador 3D a través de una NUI innovadora utilizando al Kinect como dispositivo de captura de movimiento para la interacción usuario-computadora.

El software Unity nos sirvió para crear contenido 3D interactivo, el cual es programable mediante librerías en el lenguaje de programación C# utilizando el SDK Open Natural Interaction (OpenNI) [9].

Microsoft Research invirtió veinte años de desarrollo en la tecnología del Kinect, la Figura 1 muestra las partes que constituyen el Kinect. Fue anunciado por primera vez el 1 de junio de 2009 en la Electronic Entertainment Expo 2009 (E3 2009) como "Project Natal", creado por Alex Kipman, desarrollado por Microsoft para la consola de videojuegos Xbox 360, y lanzado en Norteamérica el 4 de noviembre de 2010,

desde junio del 2011 se lanzó su versión beta del SDK para PC para el Sistema Operativo Windows 7 [10].



Fig. 1. Kinect de Microsoft.

El Kinect es un dispositivo alargado, diseñado para estar en una posición horizontal. El dispositivo contiene, una cámara RGB, sensores de profundidad, un micrófono multi-array y un motor de inclinación [11].

- Cámara RGB, es una cámara de video con una resolución de 640x480 píxeles a 30 Marcos Por Segundo (Frames Per Second, FPS por sus siglas en inglés).
- Sensores de profundidad, es una combinación de un proyector de profundidad (retícula izquierda) con un sensor de profundidad (retícula derecha), se calcula la distancia en función del tiempo que tarda en reflejar la luz.
- Micrófono Multi-array, es un conjunto de cuatro micrófonos que se monta como un solo micrófono.
- Motor de inclinación, nos permite ajustar el Kinect hacia arriba o hacia abajo hasta 27°.

Y aunque no visibles a simple vista, Kinect también posee:

- Memoria RAM de 512 Mb.
- Acelerómetro, para estabilizar la imagen cuando se mueve.
- Ventilador, no está encendido continuamente para no interferir con los micrófonos.

En conjunto nos permite capturar imágenes y movimientos de los cuerpos, además de ofrecer reconocimiento facial y aceptar comandos de voz.

La intención de utilizar el Kinect con respecto a las distintas aplicaciones computacionales tradicionales está enfocada en reemplazar la entrada de teclado y ratón estándar por gestos basados en umbrales angulares y la distancia entre las partes del cuerpo.

La idea de usar el cuerpo como medio de interacción con las aplicaciones computacionales dio pauta a la NUI, como sucesora de la GUI. La NUI se basa en el uso de habilidades del cuerpo, tales como movimientos y gesticulaciones, para ser utilizadas como medio de interacción con las computadoras, eliminando el uso de los dispositivos de entrada estándar. Es decir, el objetivo de las NUI es lograr que el usuario interactúe con la computadora de una más natural a como lo hacen las personas, sin que el usuario esté consciente de utilizar un dispositivo de rastreo de movimiento y voz. El desarrollo del Kinect ha constituido a la consola Xbox 360 de Microsoft como la primer NUI comercial [12].

2. Estado del arte

Las interfaces de usuario fueron creadas con el propósito de facilitar la interacción hombre-máquina. Así, la evolución de la computación ayudo también a las interfaces de usuario haciendolas cada vez más poderosas, las cuales buscan ser cada vez más naturales y amigables, adaptándose a las necesidades del usuario. En la última década un nuevo tipo de interfaces de usuario, denominadas NUI, han comenzado a ganar terreno sobre el paradigma GUI, que ha sido el más exitoso y popular para las interfaces de usuario.

En la realización del presente trabajo se consultaron diversos artículos y empresas que están relacionadas con aplicaciones que utilizan NUI mediante el Kinect con la finalidad de eliminar el uso del ratón y teclado para llevar a cabo una forma de interacción más natural con la computadora. Algunos de los artículos y empresas analizadas, además de controlar una aplicación, se enfocan en el análisis de los datos obtenidos con Kinect, o su integración con otros dispositivos referentes al área de investigación.

2.1 Controlar una aplicación

KinEmote [13]: Controla el ratón con la mano. En este trabajo se muestra una aplicación para Windows que permite controlar el ratón con la mano. Está basada en OpenNI/NITE y el reconocimiento de gestos con la mano: juntar el dedo índice y pulgar para hacer clic izquierdo, cerrar la mano para arrastrar y la palma de la mano para deslizarse por las barras de desplazamiento de las aplicaciones.

Kinect-Education [14]: Kinect-Education, es una empresa ya dedicada al desarrollo de software de aprendizaje para niños implementando el uso del Kinect.

Análisis de datos y otros dispositivos: A Kinect-based system for physical rehabilitation. A pilot study for young adults with motor disabilities [15]: En este artículo se muestra una aplicación que ayuda a los pacientes con necesidad de realizar terapias

para recuperación, además de ser una aplicación muy llamativa da la posibilidad de realizar los ejercicios desde casa ofreciendo también una retroalimentación a quien los practican.

Skaneect [16]: Sirve para modelar en 3D, el entorno donde se encuentra el usuario con solo girar poco a poco el Kinect para que escanee el lugar, también puede escanear el cuerpo de una persona y así tener la posibilidad de reconstruir e implementar un personaje en alguna aplicación como un videojuego con un avatar igual al usuario.

Como se puede ver en los distintos trabajos del estado del arte presentados anteriormente, todos utilizan una NUI, que hacen uso de los movimientos y gesticulaciones del cuerpo humano como datos de entrada, así el usuario puede emplear movimientos que le parecen familiares para crear una interacción más natural con la computadora, como si estuviera manejando objetos reales.

3. Metodología usada

El presente trabajo presenta una propuesta para la Manipulación de un Navegador 3D por medio del software Unity que sirve para modelar en 3D, mediante gestos y reconocimiento de voz utilizando el SDK OpenNI y Kinect de Microsoft.

La inicialización se realiza con la conexión del Kinect y el Unity, posteriormente se inicia una fase de funcionamiento que se detiene hasta que se indique el cierre del sistema. La fase comienza por obtener las imágenes de video, profundidad y el esqueleto del usuario.

Cámara de video: El Kinect adquiere imágenes de video con un sensor CMOS de colores a una frecuencia de 30 Hz, en colores RGB de 32-bits y resolución VGA de 640×480 píxeles. El canal de video monocromo CMOS es de 16-bit, resolución QVGA de 320×240 píxeles con hasta 65,536 niveles de sensibilidad. En un amplio campo visual con objetos, la cámara de video trata de reconocer a qué distancia están los distintos objetos, distinguiendo movimientos en tiempo real.

Sensores de profundidad: Para calcular distancias entre un cuerpo y el sensor, el sensor emite un haz láser infrarrojo que proyecta un patrón de puntos sobre los cuerpos cuya distancia se determina. Una cámara infrarroja capta este patrón y por hardware calcula la profundidad de cada punto. El rango de profundidad del sensor de Kinect está entre 0.4 y 4 metros. Existen 2 modos (Default y Near) para determinar distancias. Se ha elegido el modo "Default" ya que permite medir hasta 4 metros de distancia con respecto al sensor. Los sensores pueden llegar a distinguir la profundidad de cada objeto con una resolución de 1 centímetro y las estimaciones de la altura y anchura con una exactitud de aproximadamente 3 milímetros.

3.1 A partir del esqueleto obtenido se realiza la detección de acciones

La aplicación funciona a través del Kinect, detectando los movimientos de la persona que interactúe con el dispositivo y a su vez la aplicación transforma estos movimientos en instrucciones. La aplicación detecta la posición de las manos para poder mover o acercar un objeto. También es posible realizar un recorrido a través de un mapa tridimensional mediante la detección del movimiento corporal del usuario.

3.2 Interfaz Gráfica

La interfaz gráfica de nuestro navegador 3D propuesto se muestra en la Figura 2. El diseño de la interfaz del usuario es minimalista para que el usuario pueda interactuar con los objetos 3D de una manera sencilla por medio de nuestra NUI utilizando el Kinect.

En la Figura 2, se puede ver un prototipo del menú de la interfaz gráfica del navegador 3D, las opciones de los objetos Corazón y Oreja tendrán una animación de carrusel, el usuario puede elegir la opción deseada cuando se posicione sobre ella con la pelota, que es quien sigue la posición de la mano.



Figura 2. Interfaz gráfica del Navegador 3D.

Cuando el usuario posicione la pelota sobre la opción “Corazón”, la aplicación nos redireccionará a la escena Corazón mostrada en la Figura 3. El cual es una vista de un corazón, donde el usuario podrá manipular el objeto con el Kinect, y las funcionalidades son: acercar, alejar y girar.

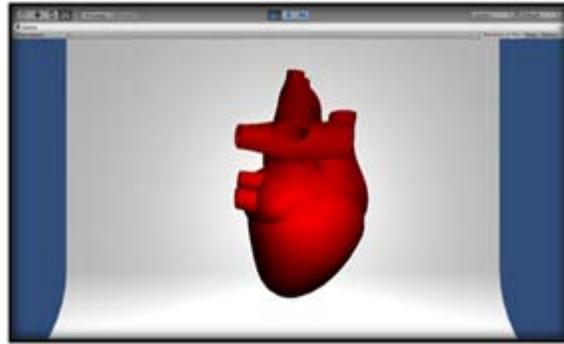


Figura 3. Vista de la escena: Corazón.

De la misma forma cuando el usuario posicione la pelota sobre la opción “Oreja”, la aplicación nos redireccionará a la escena Oreja, mostrada en la Figura 4. La cual es la vista de una oreja, donde el usuario podrá manipular el objeto con el Kinect, y las funcionalidades son: acercar, alejar y girar.



Fig. 4. Vista de la escena: Oreja.

3.3 Patrón Modelo-Vista-Controlador

El patrón Modelo-Vista-Controlador (Model-View-Controller, MVC por sus siglas en inglés) es un patrón compuesto para desarrollar aplicaciones complejas, que fue utilizado en la arquitectura de nuestra propuesta. El patrón tiene tres bloques: Modelo, Vista y Controlador. El Modelo tiene los datos de la aplicación, y la lógica de ne-

gocios de nuestra aplicación, la Vista representa la GUI hacia el usuario y muestra el estado de la aplicación, y el Controlador maneja las entradas del usuario y cambia los estados de la aplicación. El punto fundamental del patrón es la alta flexibilidad del patrón claramente aplicable a la separación de los tres bloques, sin superponer sus responsabilidades, posibilitando a cada bloque llevarlas a cabo, pero estos bloques colaboran de forma vinculada, comunicándose entre ellos. La Figura 3 muestra los patrones: MVC, Observador y Singleton, del núcleo fundamental de nuestra arquitectura. El patrón Observador nos posibilito mantener actualizadas las Vistas con el Modelo. El patrón Singleton se centra en asegurar que sólo haya una instancia de una clase en la memoria, y sólo se tenga un acceso global al objeto, con el objetivo de optimizar el uso de los recursos, en nuestra propuesta el patrón Singleton optimiza el acceso a la capa del Modelo. Estos patrones ya han sido utilizados en el desarrollo de propuestas previas [17].

3.4 Rastreo de esqueleto y técnicas de definición algorítmica de los gestos:

El Rastreo del Esqueleto (Skeleton Tracking, ST por sus siglas en inglés) lo basamos en un algoritmo de reconocimiento, el algoritmo tiene un entrenamiento basado en muchas imágenes, con la finalidad de lograr una buena precisión en la identificación de los esqueletos [18]. Nuestra propuesta implementa el ST en varias fases. La primera fase captura datos del mapa de profundidad, en la segunda fase se hace una clasificación de las partes del cuerpo con la finalidad de conseguir sus articulaciones, la tercera fase construye el esqueleto en función de las articulaciones.

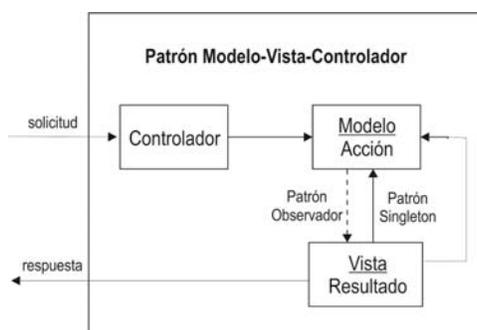


Fig. 5. Patrón Modelo-Vista-Controlador.

En nuestra propuesta implementamos diferentes técnicas para definir algorítmicamente los gestos, contrastamos un conjunto de plantillas preestablecidas, comparando cada captura con las plantillas con cierto margen de error [18].

4. Conclusiones y trabajo futuro

Nuestra propuesta de una aplicación de un navegador 3D con una NUI innovadora, para manipular objetos 3D puede ser de gran utilidad en el área de la educación al estimular el aprendizaje por medio de simulaciones interactivas eliminando el factor miedo al momento de aplicar la metodología de prueba y error cuando se utiliza instrumental de alto riesgo o costoso. Este tipo de desarrollos posibilitan agregar herramientas innovadoras en los salones de clase, con la finalidad de estimular la investigación de los estudiantes, y mejorar su entendimiento de los temas tratados.

Las simulaciones computacionales hacen los conceptos más atrayentes y menos genéricos, posibilitando a los estudiantes una mejor comprensión y desarrollar nuevas habilidades, mejorando un entendimiento realizado por medio de la tecnología. Las computadoras mejoran las actitudes e intereses de los estudiantes, al posibilitar un aprendizaje personalizable en un ambiente más interactivo por medio de la NUI propuesta y agradable.

Por lo tanto, nuestra propuesta del navegador 3D puede apoyar en la solución del problema de las prácticas inaccesibles para los alumnos o para aquellas personas que quieren manejar o experimentar con materiales muy caros antes de implementar físicamente. Esta aplicación tiene la ventaja de que las personas interesadas pueden trabajar con ella desde casa con solo contar con un Kinect sin necesidad de gastar en el material para realizar una práctica, así los estudiantes podrán tener un mayor control de responsabilidad sobre sus habilidades, sobre el ¿Qué quiero aprender hoy?, y así comenzar a realizar actos de proactividad.

La propuesta pretende fortalecer el área del conocimiento en los alumnos, aprendiendo teoría pero con la opción de aterrizarlo con la práctica.

La propuesta presentada puede llegar a ser una gran herramienta de apoyo para las instituciones y sus alumnos, fomentando el gusto por sus estudios a través de las prácticas virtuales realizadas con sus propios movimientos y a su propio ritmo, sin temor a equivocarse puesto que se trata de un ambiente virtual, donde no se corre el riesgo de causar una pérdida fuerte, teniendo la opción de repetir los ejercicios un sinnúmero de veces, sin los costos generados por el uso de los materiales involucrados en las prácticas.

En el futuro cercano se crearan NUI en la mayoría de las aplicaciones computacionales cuyo objetivo será lograr que el usuario interactúe con la computadora, como si esta fuese otra persona y sin que esté consciente de utilizar un dispositivo.

Al realizar la investigación de esta propuesta se observó que el manejo de objetos 3D con el dispositivo Kinect, es un gran avance en la tecnología, que pueden implementar diferentes conceptos para: negocios, artes, educación, videos juegos, y más allá.

Como trabajo futuro se realizara un sistema que utilice una NUI para modelar o deformar objetos virtuales y experimentar con la interacción en un ambiente de realidad aumentada. En la investigación de esta propuesta, se mostró que hay muchas posibilidades en distintos campos de aplicación de utilizar el Kinect para NUIs de manera innovadora.

Agradecimientos.

Los autores de este artículo agradecen a la Universidad Politécnica de Querétaro y al Instituto Politécnico Nacional (IPN) y a la Escuela superior de Cómputo (ESCOM) por su apoyo para este trabajo dentro del proyecto SIP: 20131560. Los autores desean reconocer a todos sus colegas y estudiantes que participaron en el diseño y desarrollo del software descritos en este artículo.

Referencias

1. NUI: Interfaz de Usuario Natural, Sitio Web, Disponible en URL: <http://www.microsoft.com/en-us/kinectforwindows/develop/>.
2. Wii Official Site at Nintendo, Sitio Web, Disponible en URL: <http://www.nintendo.com/wii>.
3. Global Weekly Video Game Charts, Games Sales, Top Sellers, Game Data - VGChartz, Disponible en URL: <http://www.vgchartz.com/>.
4. PlayStation®Move Motion Controller - PlayStation®3 Move Info, Games & Updates, Disponible en URL: <http://us.playstation.com/ps3/playstation-move/>.
5. Kinect - Xbox.com, Disponible en URL: <http://www.xbox.com/es-MX/Kinect>.
6. Kinect for Windows | Voice, Movement & Gesture Recognition Technology, Disponible en URL: <http://www.microsoft.com/en-us/kinectforwindows/>.
7. Microsoft, Sitio Web, Disponible en URL: <http://www.microsoft.com/es-mx/default.aspx>.
8. Unity Technologies, Sitio Web, Disponible en URL: <http://unity3d.com/>.
9. OpenNI | The standard framework for 3D sensing, Disponible en URL: <http://www.openni.org/>.
10. J. Webb, J. Ashley, Beginning Kinect Programming with the Microsoft Kinect SDK (Apress, 2012).
11. R. Miles, Start Here! Learn the Kinect API (Microsoft, 2012).
12. S. Peralta. Interfaz de lenguaje natural usando Kinect. Tesis de Maestría en Ciencias de la computación, CINVESTAV-IPN, México, 2012.
13. KinEmote, Sitio Web, Disponible en URL: <http://www.kinemote.net/>.
14. Kinect–Education, Sitio Web, Disponible en URL: <http://www.kinecteducation.com/>.
15. Y. Changa, S. Chenb, J. Huangc. A Kinect-based system for physical rehabilitation, a pilot study for young adults with motor disabilities. Elsevier, Vol. 32, No. 6, November–December 2011, pp. 2566–2570.
16. Skanect, Sitio Web, Disponible en URL: <http://skanect.manctl.com/>.
17. Rubén Peredo, Alejandro Canales, Alain Menchaca, Iván Peredo, Intelligent Web-based education system for adaptive learning, Expert Systems with Applications, 38(12): pp. 14690–14702, Pergamon Press, 2011.
Reto SDK Kinect: Reconocer gestos con Skeletal tracking – MSDN España, Sitio Web, Disponible en URL: <http://blogs.msdn.com/b/esmsdn/archive/2011/08/22/reto-sdk-kinect-reconocer-gestos-con-skeletal-tracking.aspx>.

Administración de procesos de reingeniería de software orientada a aspectos

Isabel Espinoza-Espinoza, Ulises Juárez-Martínez

División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Orizaba
isabel.espinoza83@gmail.com, ujuarez@ito-depi.edu.mx

Resumen. Actualmente existen muchos sistemas heredados obsoletos y con un arduo trabajo de mantenimiento para los cuales es necesario reconstruirlos o desecharlos. Ante la necesidad de migrar un sistema a un nuevo enfoque, es factible aplicar un proceso de reingeniería que permita obtener un sistema que satisfaga los requerimientos y que se realice bajo los enfoques y paradigmas actuales que mejoran la calidad del producto. Recientes enfoques de programación otorgan grandes beneficios, como lo es el paradigma orientado a aspectos (OA), éste provee un mejor análisis y comprensión del sistema, mayor nivel de abstracción, mayor legibilidad en el código, alto nivel de reutilización, así como también facilita la adaptación y el mantenimiento. Es de suma importancia administrar adecuadamente los procesos de reingeniería para que el nuevo sistema cumpla con requerimientos de calidad. Bajo dichas premisas este trabajo presenta una propuesta de administración de procesos de reingeniería OA alineada a MoProSoft.

Palabras clave: Reingeniería de software, orientación a aspectos, desarrollo de software basado en componentes, MoProSoft.

1. Introducción

Ante la creciente necesidad de ser más competitivos en la industria del software, especialmente con estrategias que promuevan su reutilización y escalabilidad, es necesario no sólo transferir conocimiento de nuevos paradigmas, además es imperante que las empresas de desarrollo de software cuenten con elementos tangibles de casos que les permitan formar una base de insumos reutilizables con el objetivo de incrementar su productividad. Con base en la amplia literatura reportada, en términos generales, la capacidad de adaptación y mantenimiento de los sistemas de software, la reutilización de componentes, así como la evolución en general, se ve favorecida con el modelo OA. La problemática esencial radica en cómo aplicar actividades de mantenimiento a insumos y/o productos de software que soporten las mejores prácticas de adaptación. Dado que la OA favorece este tipo de actividades, se plantea una propuesta de administración de procesos de reingeniería de software que permita lograr principalmente tres atributos de calidad de los sistemas en general: la adaptación, mejorar

el mantenimiento y robustecer la escalabilidad; además de favorecer la adopción del paradigma OA en la industria.

El presente trabajo está alineado a la iniciativa de mejora continua en la industria de desarrollo de software, consiste en el análisis y selección de técnicas, métodos y herramientas óptimas para la aplicación de estrategias de reingeniería de software OA. Con dicha finalidad se ha tomado un caso de estudio real de la industria (sección 4), actualmente en operación, el cual requiere alta capacidad de adaptación y mantenimiento pero al mismo tiempo presenta problemas fuertes de código invasivo y disperso. Los insumos, así como el producto final, deben ser adaptables ante las diferentes necesidades, siendo regulado con lineamientos fiscales cada vez que la institución acreditada lo solicite. Ante esta situación de adaptación, se requiere una solución independiente dedicada exclusivamente al proceso de negocio en cuestión, que sea modular y de fácil mantenimiento, con poca afectación en funcionalidad en los cambios que pudieran surgir tanto en la propia solución, como en otra que lo utilice. También es deseable evitar el volver a ejecutar pruebas a toda la funcionalidad, o causar nuevos errores al realizar cambios.

Este artículo está organizado como se indica a continuación. La sección 2 presenta los antecedentes, la sección 3 muestra los trabajos relacionados así como el análisis de las estrategias de reingeniería OA, el caso de estudio se presenta en la sección 4. A continuación, en la sección 5 se presenta la metodología enfocada en la administración de procesos de reingeniería OA y la discusión de la misma en la sección 6. Finalmente se dan las conclusiones y el trabajo a futuro en las secciones 7 y 8 respectivamente.

2. Antecedentes

Este trabajo se fundamenta en cinco elementos principales que se exponen a continuación.

2.1 Reingeniería de software

De acuerdo a [1] las tareas de reingeniería se hacen presentes cuando se requieren cambios en un sistema de software existente con la intención de cambiar su estructura o fundamento técnico, tales como dependencias del sistema operativo, al necesitarse cambios en: la plataforma, bases de datos y procesamiento de transacciones del sistema. También se necesita cuando hay documentación incompleta, baja comprensión de código fuente o dependencias complicadas entre los módulos. Al mismo tiempo se da cuando la tecnología se vuelve obsoleta o ya no es la deseada, de tal forma que resulta necesario realizar una migración como consecuencia del cambio de lenguaje de programación. El proceso de reingeniería consiste en reconstruir un sistema, creando un producto con una funcionalidad nueva, un mejor rendimiento y fiabilidad, y un mantenimiento mejorado.

Considerando que la OA aporta mayores beneficios en el desarrollo de software, un proceso de reingeniería de software OA debe permitir al mismo tiempo la detección de aspectos.

2.2 Desarrollo de software OA

El desarrollo de software OA tiene como objetivo hacer los sistemas más fáciles de mantener y reutilizar al ofrecer mecanismos para encapsular requerimientos no funcionales [2]. Este desarrollo está centrado en la separación de asuntos (concerns); un asunto es algo de interés para un stakeholder o para un grupo de ellos (tales como asuntos funcionales, de calidad de servicios, de políticas, organizacionales, entre otros). En [2] se establecen las siguientes actividades para el desarrollo de software OA:

- Ingeniería de requerimientos orientada a asuntos, en la que se identifica el conjunto de requerimientos funcionales y no funcionales del sistema.
- En el diseño se identifican y diseñan los aspectos, especificando los lugares dónde éstos se entrelazan con las funcionalidades del sistema (figura 1).
- En la fase de implementación se codifican las funcionalidades centrales y los aspectos.
- La fase de verificación y validación busca demostrar a través de un conjunto de pruebas que el sistema reúne las especificaciones y las necesidades reales de los stakeholders.

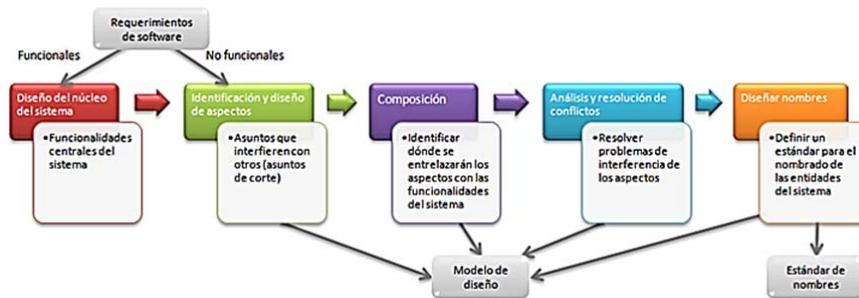


Figura 1. Proceso de diseño OA, adaptado de [2].

A diferencia de un desarrollo de software OA, en la fase de mantenimiento del desarrollo de software Orientado a Objetos (OO) cuando los requerimientos iniciales cambian, se invierte mucho tiempo analizando cuáles partes de los componentes cambian, incluso se omiten algunas secciones que tienen que cambiar, o en otro caso se introducen errores en el sistema, además existe la posibilidad de introducir código

enredado (tangling) o disperso (scattering); el primero se presenta cuando un módulo en un sistema incluye código que se implementa en diferentes requerimientos del sistema; el segundo se da cuando la implementación de un asunto se dispersa a través de muchos componentes en un programa. Al desarrollar software OA, los aspectos se comprenden, reutilizan y modifican de forma independiente, contrarrestando la situación anterior.

2.3 Programación orientada a aspectos

La programación OA es una técnica que permite la abstracción y encapsulación de asuntos de corte (aspectos) proporcionando nuevos niveles de modularidad. Trabaja sobre paradigmas existentes (estructurado y OO) gracias al entrelazado se da la interacción entre aspectos y objetos, brindando así la infraestructura necesaria para producir sistemas completos.

Los lenguajes OA trabajan con: eventos identificables en la ejecución de un programa (para clases, métodos y campos) llamados puntos de unión, conjuntos de puntos de unión conocidos como cortes y avisos que son el código que se ejecuta al identificar un punto de unión, los avisos son módulos reutilizables. Adicionalmente los lenguajes OA soportan asuntos identificables como logging, tracing, profiling, aplicación de políticas, optimización, seguridad, autenticación, autorización y administración de transacciones. La evolución de un sistema se da de forma estática al modificar la estructura de las clases y dinámica al detectar eventos en tiempo de ejecución y ejecuciones de métodos y acceso a campos.

2.4 Ingeniería de software basada en componentes

Considerando que el caso de estudio dará como resultado un software que se integre a los sistemas de los clientes o a un ERP de la empresa, surge la necesidad de emplear un desarrollo de software basado en componentes. La ingeniería de software basada en componentes es el proceso bajo el cual: se definen, implementan, integran o componen dentro de un sistema, un conjunto de componentes independientes con bajo acoplamiento y alta cohesión [2].

Los elementos esenciales que cumplirá un sistema de software basado en componentes son: independencia, estar contruidos bajo un estándar, contar con un middleware que brinde soporte a la integración de componentes y realizarse bajo un proceso de desarrollo orientado hacia la ingeniería de software basada en componentes.

Los componentes al ser independientes no interfieren con otras operaciones, ocultan los detalles de implementación, por lo que si la implementación de algún componente cambia, no se afecta el resto del sistema. La comunicación entre componentes se da por interfaces bien definidas, de tal forma que si se le da mantenimiento a una interfaz, un componente será reemplazado por otro que brinde funcionalidades mejoradas o adicionales en caso de que sea necesario el replazo. La infraestructura de los

componentes provee una plataforma de alto nivel que reduce los costos en el desarrollo de aplicaciones.

2.5 Administración de proyectos específicos

MoProSoft es un modelo de procesos para la industria de software mexicano, su objetivo es fomentar la estandarización incorporando las mejores prácticas en gestión e ingeniería de software, brindando un modelo basado en estándares internacionales que es fácil de entender y de aplicar, no es costoso en su adopción y busca ser la base para alcanzar evaluaciones exitosas con otros modelos y normas como ISO 9000, 2000 o CMM V 1.1 [3].

La estructura de procesos de MoProSoft dentro de la categoría Operación cuenta con un proceso llamado Administración de proyectos específicos, tiene como propósito establecer y llevar a cabo sistemáticamente las actividades que logren cumplir con los objetivos de un proyecto en tiempo y costo esperados [3].

Sus objetivos son:

- Lograr los objetivos del proyecto en tiempo y costo a través de la coordinación y el manejo de recursos.
- Mantener informado al cliente a través de reuniones de avance del proyecto.
- Atender las solicitudes de cambio del cliente mediante la recepción y análisis de estas.

El logro de los objetivos mencionados estará dado por la realización de las actividades mostradas en la figura 2, las cuales son incorporadas dentro del presente trabajo con el fin de administrar adecuadamente los procesos de reingeniería OA.



Figura 2. Administración de proyectos específicos.

3. Trabajos relacionados

En [4] se plantea que algunas veces muchos documentos y código fuente se pierde, conservando solo los archivos binarios ejecutables, los cuales no se pueden actualizar sin documentos o código fuente que sustente los cambios. Para ello se realizó un trabajo de reingeniería a sistemas heredados que contaban solo con el archivo de código binario ejecutable, utilizando para ello técnicas de reflexión y descompilación para generar diagramas de clase.

En [5] se aborda el modelo de diseño para la orientación a aspectos (DMAsp Design Model for Aspect Orientation), el cual se basa en el enfoque de aspectos y tiene soporte computacional de la ReJAsp (Reingeniería con Java usando AspectJ). Considera tres fases: entender la funcionalidad del software, abordar los asuntos de corte y comparar el software OO con el software OA.

En [6] se plantean técnicas como la identificación de interfaces candidatas, minería de aspectos, minería combinada, análisis identificado y recomposición, tales técnicas para llevar a cabo la migración hacia aspectos. El estudio indica que la migración de interfaces candidatas a aspectos tiene un impacto limitado en el tamaño de la descomposición principal, pero al mismo tiempo produce una mejora en el código modularizado. Desde el punto de vista de atributos de calidad externos, la modularización de la implementación de las interfaces de cortes en punto simplifica la comprensión del código fuente.

En [7] se hace hincapié en la dificultad de identificar y comprender cuáles son las porciones de código directa o indirectamente afectadas por aspectos y cuáles se ejecutan en realidad en un punto dado en el código, para ello se propone un grafo de flujo de control de aspectos inter procedimentales, que representa las interacciones entre los aspectos y los componentes OO. También se utiliza un modelo métrico para evaluar el nivel de interferencia que los aspectos pueden introducir en un sistema OA.

En [8] se presenta un enfoque de ingeniería inversa a Interfaz Gráfica de Usuario (GUI) que resulta innovador gracias a su uso eficiente, la instrumentación no intrusiva y la técnica de análisis dinámico; se utiliza AspectJ para monitorizar y registrar la información de la GUI de una aplicación Java.

En [9] se menciona que la mayoría del software existente no fue diseñado para adaptación, por ello se propuso un enfoque basado en modelos para presentar la adaptación dinámica a los sistemas heredados no adaptativos, mientras se garantizan sus propiedades de mantenimiento.

En [10] se describe la reingeniería de una biblioteca OO llamada Framework de edición gráfica. Se encontraron mejoras en la modularidad pero se observó la necesidad de herramientas de automatización y recomposición para apoyar la reingeniería OA. Los resultados indicaron la reducción de la complejidad y del acoplamiento, pero sólo a un pequeño grado, se asume que el resultado estuvo limitado al corto tiempo de ejecución.

3.1 Análisis de estrategias de reingeniería OA

La tabla 1 muestra un análisis comparativo de las técnicas mencionadas previamente que los autores proponen utilizar para la resolución de problemas similares al del caso de estudio en cuestión.

Tabla 1. Técnicas de reingeniería OA relacionadas con el proyecto

Técnica	Objetivo	Ventaja	Desventaja
Identificación de interfaces candidatas a aspectos [6]	Identificación de aspectos a partir de: <ul style="list-style-type: none"> • Interfaces de paquetes externos. • Relación de nombres de las interfaces implementadas. • Agrupamiento de métodos de acuerdo a la relación de llamadas. 	Probado en proyectos con más de medio millón de líneas de código. Mejora la comprensión de código, mantenimiento y modularidad.	Deben existir interfaces bien definidas e implementadas, así como seguir estándares de nomenclatura. Impacto limitado en la reducción del tamaño del código.
Análisis identificativo [6]	Identificación de asuntos de corte a través de la nomenclatura.		
Recomposición [6]	Conversión de interfaces candidatas en aspectos.		
Modelo métrico [7]	Evalúa la interferencia que los aspectos introducen en un sistema OA.	Se analiza la alteración estática o dinámica en los componentes base a partir de la interferencia de los aspectos. Considerando la magnitud de interferencia se decide si los componentes están bien diseñados, si necesitan ser reestructurados o se les debe aplicar reingeniería.	Se necesita una herramienta automatizada para evaluar el grafo de flujo de control de aspectos interprocedimentales con base a los criterios propuestos.
Modelo de conducción para solventar la adaptación dinámica de sistemas heredados no adaptativos [9]	Aseguramiento de la adaptación dinámica a través de: Análisis de requerimientos: selección de las propiedades que necesita satisfacer el	Modelo simple.	No presenta técnicas innovadoras.

Técnica	Objetivo	Ventaja	Desventaja
	<p>software.</p> <p>Análisis y diseño del modelo: se generan diagramas de estado a partir del código fuente y se verifican. Se crea y verifica un modelo de adaptación.</p> <p>Implementación: usa clases en Java que implementen el comportamiento requerido en el modelo de adaptación, se utiliza OA para incorporar estas clases y su implementación.</p>		
<p>Recomposición OA de forma semi automática [10]</p>	<p>Recomposición de código usando renombre, composición, relocalización, descomposición y abstracción de elementos tales como identificadores, métodos y clases. Extrae características en aspectos y parte de código en avisos.</p>	<p>Los resultados basados en las métricas aplicadas indican que se redujo la complejidad y el acoplamiento.</p>	<p>Algunos autores obtuvieron resultados desfavorables en la aplicación de recomposición:</p> <p>Reducción de 10% de código sólo después de 7 meses de recomposición (código de 140 Kloc de Java).</p>
<p>Métricas empleadas [10]:</p> <p>Aopmetrisc (herramienta para código OO y OA).</p> <p>Tamaño (LOC).</p> <p>Complejidad de operaciones por módulo.</p> <p>Acoplamiento en la llamada de métodos.</p> <p>Falta de cohesión en operaciones.</p> <p>Dependencia de paquetes.</p> <p>OA: verifica correspondencia de asuntos de corte con aspectos.</p>	<p>Obtiene indicadores de la calidad final del sistema al que se le aplicó recomposición de código.</p> <p>Se uso AJDT en eclipse y JUnit para pruebas de unidad.</p>		<p>En sistemas donde se modifica el código de 20 a 25% esto puede ser contraproducente.</p> <p>Gran cantidad de tiempo invertido en el análisis y rediseño, sobre todo al identificar aspectos.</p>

4. Caso de estudio

El caso de estudio se centra en el proceso de negocio de facturación electrónica, está desarrollado bajo el paradigma OO, en el sistema operativo Windows 7 utilizando el lenguaje C#.

Actualmente existen dos versiones del producto desarrollado, para ellas se requirieron actualizaciones en los requerimientos solicitados por la institución que establece lineamientos y políticas para dicho producto de software, sin embargo, existe documentación incompleta e inconsistente para fundamentar dicho trabajo. Nuestro caso de estudio corresponde a la tercera versión del producto de software, el cual considera todas las actualizaciones pertinentes.

En el análisis de información del sistema actual, se ha detectado la ausencia del diagrama de clases cuya vista estática es de suma importancia para entender en términos de objetos cómo está constituido el sistema y cómo se relacionan sus elementos, también hay inconsistencias en los esquemas de bases de datos respecto a los cambios requeridos recientemente por la institución reguladora.

En el análisis del código fuente, se identificaron casos de código invasivo y disperso. Por ejemplo, el código de la figura 3 corresponde a la incorporación de impuestos trasladados a un comprobante fiscal. El código invasivo se observa en las líneas 4 a la 8, debido a que el impuesto trasladado está fijo, ocasionando fuertes problemas de mantenimiento. Ante un cambio deben actualizarse directamente las líneas de código correspondientes en todos los módulos involucrados.

En el código fuente también se observó que los componentes que conforman el sistema actual han sido desarrollados como “paquetes” que agrupan clases relacionadas por la funcionalidad que les corresponde, sin embargo, algunos componentes son altamente acoplados y bajamente cohesivos al depender entre sí y no modularizarse adecuadamente.

```

1 List<AdesoftSerializador.CFD32.ComprobanteImpuestosTraslado> MyTraslado=
2 new List<AdesoftSerializador.CFD32.ComprobanteImpuestosTraslado> ();
3 AdesoftSerializador.CFD32.ComprobanteImpuestosTraslado MyIvaT=
4 new AdesoftSerializador.CFD32.ComprobanteImpuestosTraslado () {
5     importe=decimal.Parse ("16.00"),
6     impuesto=AdesoftSerializador.CFD32.ComprobanteImpuestosTraslado.IVA,
7     tasa=decimal.Parse ("16.000000")
8 };
9 Mytraslado.Add(MyIvaT);

```

Figura 3. Código invasivo

Considerando lo anterior, se ha identificado el conjunto de actividades necesarias para proponer una metodología que brinde los lineamientos a seguir para un proceso de reingeniería OA, para ello se han analizado las técnicas OA (sección 3.1) incorporando aquellas que son viables a este proyecto.

Dicha metodología considera la concepción del nuevo sistema identificando, separando y encapsulando los asuntos de corte, a fin de tener un código no disperso, no invasivo, entendible y reutilizable, con ello será más fácil y eficiente su mantenimiento pese a los cambios que se presenten, facilitando así la evolución del sistema. Así

también se considera la modularización adecuada de las funcionalidades del sistema a fin de producir componentes independientes y cohesivos.

La metodología incluye la documentación de artefactos de software OA para la reingeniería: ingeniería inversa, arquitectura, análisis, diseño e implementación.

5. Metodología propuesta

A diferencia de otras aproximaciones de reingeniería OA, la metodología propuesta utiliza el Enfoque de Temas [11] que indica lineamientos robustos para analizar y diseñar software OA. También se utiliza la separación de asuntos simétrica, que permite separar los aspectos de sus propias funcionalidades y de la funcionalidad central del sistema para una mejor modularidad, al final se integran los aspectos con las funcionalidades generando el sistema completo. Se ha integrado ASAAM (Método de Análisis Aspectual de Arquitectura de Software) [12] para evaluar los componentes resultantes.

Nuestra propuesta integra además una cuidadosa selección de técnicas (sección 3.1) viables para robustecer su aplicación práctica, efectiva y tangible. Incluso no se requiere de la adquisición de herramientas o el aprendizaje de técnicas sofisticadas que repercutan en el costo y tiempo de realización del proyecto a diferencia de las técnicas descartadas. Se excluyen además aquellas que son dependientes de algún lenguaje de programación.

A continuación se presenta la metodología que comprende la administración de procesos para cumplir con el propósito del caso de estudio.

Planificación

- Definición del plan del proyecto [3].
- Definición del plan de desarrollo [3].
- Definición del plan de verificación y validación [3].
- Definición de acciones correctivas y preventivas [3].

Análisis y diseño

- Actualización de los requerimientos funcionales, no funcionales y las restricciones del sistema, a partir de la información recabada del sistema actual OO y de los cambios requeridos.
- Aplicación de ingeniería inversa para la obtención de los modelos faltantes e incompletos (en nuestro caso se utiliza la herramienta CASE Enterprise Architect).
- Adecuación del modelado recuperado (paso anterior) de acuerdo al análisis y diseño OA del Enfoque de Temas [11].
- Identificación de aspectos [6] a partir del análisis del código perteneciente al sistema actual.
- Diseño de componentes de la aplicación.
- Registro del reporte de actividades [3].
- Revisión del plan del proyecto y plan de desarrollo [3].

Implementación

- Migración de código fuente (en nuestro caso de C# a Java).
- Recomposición de código fuente del sistema actual OO al detectarse código enredado, duplicado o disperso [6].
 - Codificación del producto. Se considera el lenguaje AspectJ ya que es el más utilizado en la industria, su modelo de puntos de unión es el más expresivo y robusto y es utilizado por otros lenguajes OA, adicionalmente se cuenta con mayor documentación y soporte que respalde su uso. Para trabajar con AspectJ se utilizará el IDE Eclipse.
 - Registro del reporte de actividades [3].

Pruebas

- Evaluación de los componentes construidos. Se consideran las reglas heurísticas de ASAAM para identificar si los componentes son cohesivos, compuestos, enmarañados o mal definidos. ASAAM cubre la evaluación de la interferencia que los aspectos introducen a un sistema OA sugerida por [7] al identificar la necesidad de reestructurar los componentes o aplicarles reingeniería en caso de que estos hayan sido concebidos inadecuadamente. Con ASAAM también se logran satisfacer los objetivos de las métricas empleadas por [10].

Cierre

- Actualización del reporte de actividades [3].
- Registro del seguimiento al proyecto [3].
- Evaluación del plan del proyecto [3].
- Evaluación del plan del desarrollo [3].

Las actividades correspondientes al modelo propuesto por [9] se cubren totalmente con la ejecución de la metodología propuesta.

Finalmente para comprobar la mejora en el mantenimiento, se planea la definición de cambios en el nuevo sistema OA a fin de verificar su impacto. Si surgen modificaciones por la institución reguladora del producto de software, se atenderán dichos cambios verificando la facilidad de mantenimiento, considerando para ello la cantidad de tiempo y esfuerzo invertido utilizando las métricas de COCOMO [2].

6. Discusión

La aplicación de ésta metodología guía a los interesados en la ejecución de un proceso de reingeniería OA, proporcionando una guía puntual, ordenada y sencilla.

Hasta este momento se han observado mejoras de consistencia y mantenimiento a aplicar en el sistema actual que será reconstruido; las mejoras ya son tangibles al identificar código invasivo en el sistema (sección 4) alto acoplamiento y baja cohesión en los componentes desarrollados; lo anterior indica la necesidad de completar la

ejecución de la propuesta planteada. Los resultados de este proyecto servirán para promover la adopción del paradigma OA en la industria a través de la difusión de las lecciones aprendidas durante la realización del caso de estudio.

Se observó que los problemas planteados en la literatura reportada surgieron de la necesidad de aplicar procesos de reingeniería bajo situaciones comunes como: falta de documentación, mejora de procesos, recomposición de código, entre otras; sin embargo, para la aplicación de técnicas tanto en reingeniería como en la incorporación del paradigma OA, en su mayoría los problemas se plantean con casos de estudio que fueron concebidos para comprobar la efectividad de las técnicas empleadas. Esta metodología se diferencia de las demás al contar con un caso de estudio real de la industria, que reúne las condiciones necesarias para incorporar los cuatro enfoques: las mejores prácticas de reingeniería, la promoción de la adopción del paradigma OA, el desarrollo de software basado en componentes y la administración de procesos de reingeniería alineada a MoProSoft. Todo ello generará conocimiento sobre los cuatro enfoques y permitirá la toma de decisiones principalmente para la adopción del paradigma OA en la industria de software y con ello la mejora de procesos de software y la evolución del software.

La industria de desarrollo de software se verá beneficiada por las siguientes razones:

- No es necesario invertir gran cantidad de tiempo en el aprendizaje del conjunto de técnicas y actividades planteadas.
- La curva de aprendizaje necesaria para la adopción de un nuevo enfoque dependerá de las características del equipo de trabajo, sin embargo, se asume que está no debería ser alta para personas que se dediquen al desarrollo de software.
- Este caso de estudio brinda pruebas tangibles que indican que en casos reales funciona una nueva forma de trabajo, descartando la idea de: “la teoría no siempre resulta cierta en la práctica”.
- No es necesario invertir costos elevados para ejecutar la metodología.
- La difusión de las lecciones aprendidas hará que las empresas conozcan los resultados, y por lo tanto el enfoque OA ya no resulte desconocido.

7. Conclusiones

Debido a que es deseable que las empresas de desarrollo de software generen productos escalables, que tengan una larga vida útil gracias a un mantenimiento eficaz, es deseable contar con elementos que indiquen un proceso de desarrollo de software efectivo. El principal aporte de este trabajo es la propuesta metodológica, la cual está robustecida gracias a la incorporación de estrategias viables sustentadas en el análisis de los trabajos relacionados (sección 3), contando también con el empleo de métodos, enfoques y paradigmas que resultan ser efectivos en problemas del contexto planteado. Actualmente esta metodología se ha trabajado en las fases de análisis y diseño.

Esta metodología es un elemento importante al indicar la línea a seguir para la aplicación de un proceso de reingeniería OA bajo situaciones similares, requiriendo un bajo costo y corto tiempo de aprendizaje para la ejecución de las actividades indicadas.

Finalmente este trabajo confirma la necesidad de demostrar de forma tangible los beneficios de la incorporación del paradigma OA en la industria.

8. Trabajo a futuro

Como parte de un proceso evolutivo de software, se ha trabajado en las fases de planificación, análisis y diseño planteadas en la metodología propuesta, cuyos artefactos resultantes serán refinados para proceder con la implementación. Como trabajo futuro se plantea la codificación del producto correspondiente al caso de estudio, posteriormente se utilizará ASAAM [12] para evaluar que el código final (componentes) no está enmarañado o disperso y que los aspectos existentes son los necesarios o si hubo abusos en la implementación de estos. Se realizará un comparativo que demuestre los beneficios del sistema construido contra el sistema actual, dicho comparativo será respaldado por un conjunto de pruebas como por ejemplo: el conteo de líneas de código que indicará las diferencias de tamaño de los sistemas, el análisis del registro del PSP (Proceso Personal de Software) cuya información reflejará tiempos invertidos en el proceso de desarrollo.

También se realizará la difusión de las lecciones aprendidas, las cuales servirán como guía para la adopción del paradigma OA bajo enfoques similares.

Agradecimientos

A la Dirección General de Educación Superior Tecnológica y al Consejo Nacional de Ciencia y Tecnología por su apoyo en la realización del presente trabajo de investigación.

Referencias

1. J. Borchers. Invited talk: Reengineering from a practitioner's view- a personal lesson's learned assessment. 15th European Conference on Software Maintenance and Reengineering, 2011, pp. 1-2.
2. I. Sommerville. Software Engineering. United Kingdom, Addison-Wesley, 2006.
3. H. Oktaba, C. Alquicira, A. Su, et al. Modelo de Procesos para la Industria de Software. MoProSoft. Versión 1.3, México D.F. Secretaría de Economía, 2005.

4. L. Chen, G. Wang, M. Xu and Z. Zeng. Reengineering of Java Legacy System Based on Aspect-Oriented Programming. Second International Workshop on Education Technology and Computer Science (ETCS), Wuhan, 2010, pp. 220-223.
5. H.A.X. Costa, P.A. Parreira, V. Vieira de Camargo and R.A. Dellosso. Recovering Class Models Stereotyped with Crosscutting Concerns. 16th Working Conference on Reverse Engineering, Lille, 2009, pp. 311-312.
6. M. Ceccato. Automatic Support for the Migration Towards Aspects. 12th European Conference on Software Maintenance and Reengineering, Athens, 2008, pp. 298-301.
7. M.L. Bernardi. Reverse Engineering of Aspect Oriented Systems to Support their Comprehension, Evolution, Testing and Assessment. 12th European Conference on Software Maintenance and Reengineering, Athens, 2008, pp. 290-293.
8. H. Samir and A. Kamel. Automated reverse engineering of Java graphical user interfaces for web migration. ITI 5th International Conference on Information and Communications Technology, Cairo, 2007, pp. 157-162.
9. J. Zhang and B.H.C. Cheng. Towards Re-engineering Legacy Systems for Assured Dynamic Adaptation. 29th International Conference on Software Engineering Workshops, Minneapolis, MN, 2007, pp. 10.
10. A. O'Riordan. Aspect-Oriented Reengineering of an Object-oriented Library in a Short Iteration Agile Process. *Informatics (03505596)*; Vol. 35 Issue 4, 2011, pp. 499-511, 13.
11. S. Clarke, E. Baniassad. *Aspect-Oriented Analysis and Design: The Theme Approach*. Addison-Wesley, 2005.
12. B. Tekinerdogan. ASAAM: Aspectual Software Architecture Analysis Method. Working IEEE/IFIP Conference on Software Architecture, IEEE Computer Society, 2004, pp. 5-14.

Factor de Impacto sobre artículos: propuesta de indicadores que obtengan el aporte de los artículos individuales en la comunidad científica

Jesús Vázquez-Ruiz¹, Beatriz Alejandra Olivares-Zepahua²,
María Antonieta Abud-Figueroa³

División de Estudios de Posgrado e Investigación, Instituto Tecnológico de Orizaba
jvazquezruiz@acm.org; {bolivares; mabud}@ito-depi.edu.mx

Resumen. En la actualidad hay un gran número de indicadores que miden el impacto de las revistas científicas, como el Factor de Impacto, o indicadores para medir el impacto de los investigadores como el Índice H, pero dichos indicadores realizan una medida generalizada de la revista o del investigador, sin evaluar el aporte de un artículo de forma individual; para solventar tal problemática, se presenta la propuesta de arquitectura de una aplicación para medir el impacto de los artículos de los investigadores del Instituto Tecnológico de Orizaba; la arquitectura está definida en capas: la primera para la extracción de citas de artículos e información referente a estos en fuentes como *ScienceDirect*, *IEEE Xplore*, *ACM DL* y en redes sociales como *Facebook* y *Twitter*; la segunda capa realiza el cálculo de impacto de los artículos mediante indicadores propios y la tercera capa es una aplicación Web para la visualización de los resultados.

Palabras clave: Revistas científicas, factor de impacto, indicadores, índice H.

1. Introducción

Desde mediados del siglo pasado, se estableció una medida para evaluar el factor de impacto que tiene una publicación, es decir, qué tan relevante es una revista considerada como científica; anualmente este nivel se mide y se publica por el ISI (*Institute for Scientific Information*, Instituto para la Información Científica); sin embargo, esta medida no refleja completamente la relevancia que un artículo en particular tiene en la comunidad científica o, en el caso de programas de orientación profesional, en el área empresarial correspondiente. Existen varias herramientas, la mayoría con costo, que permiten conocer y analizar el factor de impacto de las publicaciones indizadas por el ISI. También existen buscadores especializados dentro de la Web para consultar las citas que existen sobre un artículo, sin embargo, todos estos casos presentan el mismo problema: un usuario hace las búsquedas de forma individual, lo que no resulta óptimo dado el alto nivel de cambio que tiene la información.

Por lo anterior, se propone el desarrollo de una aplicación Web que, partiendo de la productividad específica de los investigadores de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Orizaba en Veracruz, México, periódicamente recolecte en la Web la información de citas en los artículos incluidos en dicha productividad, para esta recolección de citas se necesita la creación de *Crawlers*, la recolección aplica Minería de Contenido sobre sitios de fuentes fidedignas en busca de citas sobre los artículos, analizar la cita en cuestión y contabilizarla.

2. Trabajos Relacionados

Para el planteamiento de indicadores de impacto a nivel de artículo se revisaron indicadores existentes a nivel de revista y de ahí se hizo el análisis para su adaptación o la creación de nuevos en su caso, algunos de los indicadores provienen de [1], en este artículo se discute que tanto el Factor de Impacto como el Factor de Prestigio de los artículos están limitados, ya que no miden el impacto de las revistas donde se citan, para ello se proponen indicadores para medir el valor de las citas en función del impacto y/o prestigio de las revistas donde se producen dichas citas; otros indicadores analizados son de [2], ya que es importante saber cómo identificar la calidad de una revista, el investigador más prestigiado y qué artículo fue el más importante en un determinado año entre otros aspectos.

Cabe señalar que existen herramientas que obtuvieron el impacto de revistas y crearon sus propios indicadores, como es el caso de [3], proyecto que se presenta como una alternativa para medir el impacto de las publicaciones o de los investigadores; el OAS (*Open Access Statistics*, Estadísticas de Acceso Abierto) y también en [4] obtuvieron el impacto de los investigadores por medio de los SECES (*search engine count estimates*, es decir el recuento de páginas o *hits*) al utilizar motores de búsqueda como son *Google*, *Yahoo* y *MSN*. También se comprobó que una red social puede medir el impacto de una revista científica como en [5], donde trabajaron con *Twitter* contabilizando los *tweets* publicados por los usuarios sobre artículos científicos, es decir publicaciones con referencias que apuntan a artículos de *JMIR (Journal of Medical Internet Research)* y con ello saber la cantidad de atención que prestan los usuarios a un tema. En [6] se comprobó que un sitio colaborativo es capaz de medir el impacto de las revistas científicas, en dicho trabajo se busca medir el impacto de los artículos de *PubMed* (motor de búsqueda de libre acceso a la base de datos *MEDLINE* de citas y resúmenes de artículos de investigación biomédica) que son citados en *WIKIPEDIA*, por medio de sus *PMID (PubMed Unique Identifier)*, un número único asignado a cada cita de un artículo de revistas biomédicas y biológicas que recolecta *PubMed*). Las herramientas mencionadas son algunas que se realizaron para obtener el impacto de las revistas científicas pero también hay propuestas para realizar una clasificación de los artículos ya publicados, como en [7] que hace hincapié en el desconocimiento del tipo de artículo publicado en una revista, es decir, si un artículo fue solamente polémico al momento de publicarlo, si tiene una gran contribución, o si sólo cubrió una necesidad puntual. Con relación al tema de extracción de contenido, en [8] se habla de las novedades en la recolección de información en la Web al ocupar

Minería Web (minería de contenido, uso y estructura) y define la minería de contenido como “la extracción de información o texto en las páginas Web”. Para realizar dicha extracción existen herramientas como la descrita en [9], la cual elimina el ruido excesivo como es la publicidad, enlaces a otros sitios y contenido que no tiene relación con la información principal; otra herramienta se presenta en [10] y sirve para crear directorios de servicios/proveedores en la Web, o el caso de [11], la herramienta EVA (*Evolutionary Virtual Agent*, Agente Virtual Evolutivo) en su cuarta versión, trata de un agente que simula una conversación real, por medio de un cliente (*chat*) al que se le solicita información y éste le responde por texto o voz. Esta herramienta tiene 2 componentes, el primero es la interfaz con un rostro en 3D el cual, al responder, lo hace con gestos, el segundo es el motor de comportamiento (cerebro) el cual extrae pertinentemente información de la Web y la usa en el flujo de la conversación.

La Minería Web tiene, entre otras ventajas, identificar por ejemplo enlaces inválidos, en [12] se muestra la problemática derivada del gran número de URLs de artículos no válidos (eliminadas, caducadas, cambiadas de servidor) y presenta recomendaciones para disminuirla. Cabe mencionar que los artículos a los que se les calculará el impacto se encuentran en una ontología de productividad propia del programa, con información previamente verificada, para ello se investigó sobre ontologías en el contexto de artículos, en [13] se utilizan métodos para clasificar documentos automáticamente y obtener la relación entre ellos por medio de ontologías y metadatos *SÁHKE*; y finalmente en [14] se mencionan las características que conforman las ontologías y la Web Semántica.

3. Conceptos

A continuación se describen los conceptos más relevantes para el mejor entendimiento del presente documento.

- Indicadores de impacto

Los indicadores de impacto son métodos para medir la relevancia que tiene una publicación, una revista o un investigador entre otros. Estas métricas o indicadores dan una idea del éxito o fracaso del elemento a medir [2].

- Ontología

Una ontología define los términos a utilizar para describir y representar un área de conocimiento. Las ontologías las utilizan las personas, las bases de datos, y las aplicaciones que necesitan compartir un dominio de información (un dominio es simplemente un área de temática específica o un área de conocimiento, tales como la medicina, fabricación de herramientas, bienes inmuebles o productividad entre otros). Las ontologías incluyen definiciones de conceptos básicos del dominio y las relaciones entre ellos que son útiles para las computadoras, en este sentido, hacen el conocimiento reutilizable [15].

- *Crawler*

Para la extracción de información de la Web se utilizan programas llamados “arañas” (también conocidos como *Spiders*, *Crawlers* o *Bots*), dichos programas se encargan de llevar a cabo un tipo concreto de tareas, en particular es responsable de recorrer las páginas Web de Internet, descargarlas o copiarlas a la computadora local y procesarlas para la obtención de información [16].

- *Selenium*

Selenium es un marco de trabajo que permite descargar el código fuente de las páginas Web, también cuenta con funciones para extraer elementos del DOM (Modelo de Objetos del Documento) para realizar pruebas o procesamiento de información. Actualmente está disponible para *Java*, *Ruby*, *Python*, *Perl*, *.Net* y *PHP* [17].

- *Jena*

Jena es un marco de trabajo en Java para crear aplicaciones de Web Semántica, Jena proporciona un conjunto de herramientas y bibliotecas de *Java* para ayudar a desarrollar la Web semántica y enlazar datos de aplicaciones, herramientas y servidores [18].

- Protocolo OAuth

OAuth (*Open Authorization*, Autorización Abierta) es un protocolo abierto que permite la autorización segura y sencilla para acceder a datos o aplicaciones que requieren autenticación sin la necesidad de estar registrados en ellas, ya sea en aplicaciones Web, móviles o de escritorio [19].

- Scribe OAuth

Es un marco de trabajo para trabajar con el protocolo OAuth en el lenguaje *Java*, *Scribe* tiene soporte para muchos servicios populares que utilizan OAuth, tales como *Facebook*, *Twitter* y *Google* entre otros [20].

4. Propuesta

En la figura 1 se muestran las capas de la arquitectura propuesta y posteriormente se define cada una de ellas.

4.1 Extracción

La capa de extracción es responsable de identificar los artículos a los que se les calcularán los indicadores de impacto, acceder a fuentes de información para extraer

información de citas siguiendo la definición existente en una ontología de indicadores.

A continuación se muestran los elementos que conforman esta capa.

4.1.1 Fuentes de Información

Las citas de los artículos se buscan en fuentes de información confiables, como son las bibliotecas virtuales de organizaciones reconocidas; por otro lado, debido al crecimiento de las redes sociales, también se considera la revisión de dichas fuentes. A continuación se describen dichas fuentes:



Figura 1. Capas de la herramienta de impacto

ACM. La ACM es la principal organización de Ciencias Computacionales que brinda a sus miembros, y a toda la comunidad de estudiantes la oportunidad de tener a su alcance los últimos avances en tecnología. Fomenta la interacción entre estudiantes, universidades y profesionistas, que es la base para un buen desarrollo académico. La ACM tiene una biblioteca virtual (ACM Digital Library) que da acceso a artículos y libros por medio de una suscripción [21].

IEEE. La IEEE es el Instituto de Ingenieros Eléctricos y Electrónicos, una asociación técnico-profesional mundial dedicada a la generación de estándares en las áreas de Ingenierías Eléctrica y Electrónica entre otras cosas.

Su objetivo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales. Ésta cuenta con una biblioteca virtual para la distribución de libros electrónicos y artículos tanto de acceso restringido por suscripción como de acceso abierto. Dicha biblioteca virtual es IEEE Xplore [22].

ScienceDirect. ScienceDirect es una parte de *Elsevier* (*Elsevier* forma parte del *Reed Elsevier PLC Group*, una importante editorial mundial y proveedor de información), es una base de datos que ofrece artículos de más de 2.500 revistas y capítulos de más de 11.000, así como también acceso a bases de datos secundarias como *Pergamon* y *NorthHolland*. ScienceDirect facilita la colaboración, premia la innovación y acelera el propio proceso de investigación. La dirección de esta plataforma es [23].

Facebook. Esta es una red social que cuenta con 600 millones de miembros que se comunican en más de 70 idiomas [24], la visibilidad del contenido de un usuario depende de la configuración, si lo restringe o lo hace libre en la Web.

En esta red social se puede compartir enlaces (URLs) de todo tipo, ya sea audio, video o documentos; para el caso del Análisis de Sentimientos planteado para este proyecto se consideran las calificaciones aprobatorias “likes” que tiene el enlace compartido del artículo, cuántos enlaces compartidos y los comentarios tanto positivos como negativos.

Twitter. Esta es una red social que cuenta con 500 millones de miembros, dicha red permite enviar mensajes de texto plano de corta longitud, con un máximo de 140 caracteres, llamados *tweets*, que se muestran en la página principal del usuario [25].

Para la aplicación del Análisis de Sentimientos planteado se considera el número de *tweets* que tiene el artículo y sus comentarios tanto positivos como negativos.

4.1.2. Ontología de Productividad. La ontología de productividad contiene información relevante de los investigadores del Instituto Tecnológico de Orizaba, así como también todos los artículos que han elaborado cada uno de ellos, esta ontología fue desarrollada en *Protégé* [26]; cabe señalar que la herramienta de extracción partirá de esta ontología para buscar las citas de cada uno de los artículos ahí contenidos.

4.1.3. Ontología de indicadores. La ontología de indicadores, contiene toda la información acerca de los indicadores propuestos, sus nombres, fórmulas, información de las fuentes e información de las citas.

4.1.4. Crawlers o herramientas de extracción. Para elaborar la herramienta de extracción se propone trabajar con el marco de trabajo *Selenium*, ya que a pesar de hacer pruebas con *Crawlers* ya existentes como son: *JSpider Crawler*, *Jobo* y *Web Harvest* [27], las fuentes seleccionadas detectaban los procesos automáticos de estas herramientas, impidiendo la extracción de información que se necesita. Para cada una de las fuentes se cuenta con un elemento que busca el artículo y otro que extrae la información pertinente.

4.2 Controlador de cálculo

En esta segunda capa se muestra la forma en la que se calculará el impacto de los artículos, como se había mencionado existen indicadores de impacto como el “Factor de Impacto” para revistas científicas, o el “Índice de Hirsch” para investigadores,

estos indicadores se obtienen a través de las citas que tengan los artículos, pero las citas se contabilizan de forma general y surge la problemática de que el impacto no es equilibrado, por ejemplo si hubiera 2 revistas con el mismo número de citas en 2 años (el mismo impacto) ¿será lo mismo el impacto de la revista cuyas citas provienen de una revista de alto impacto que el impacto de otra cuyas citas provienen de una revista de bajo impacto?, para ello en [1] propusieron indicadores que pudieran solventar esa problemática y de cierta forma darle un valor más justo al impacto. Parecido al problema ya comentado aparece uno similar para el “Instituto Tecnológico de Orizaba”, en el caso donde un investigador tuviera 2 artículos, cada uno tiene 1 cita, la incógnita sería: ¿los 2 artículos tendrán el mismo aporte en la comunidad científica?, y la respuesta es que no, a pesar de tener el mismo número de citas se tiene que conocer el impacto de cada cita para contestar qué artículo tiene un mayor aporte o contribución, ya que no sería lo mismo que una de las citas tuviera a su vez 300 citas y la otra solo 2 citas, en este caso se reflejaría que el artículo de la cita que tiene 300 citas tiene mayor aporte que el otro, ya que está siendo parte de otra investigación la cual está provocando mayor interés y desarrollo en la comunidad científica.

También es importante comentar que las redes sociales son una gran oportunidad de dar a conocer información ya que es muy fácil que se pueda distribuir; con ello un investigador podría dar a conocer sus resultados y conocer el impacto que pueda tener gracias a la popularidad que pueda provocar. Por lo que se ha planteado sobre el impacto de artículos, se han adaptado indicadores que reflejen dicho impacto, los cuales a continuación se definen y se muestra cómo se obtienen.

4.2.1. Factor de Impacto de Artículo (FIA).

Se refiere al impacto que tiene un artículo de forma general, es decir sin importar el impacto que tengan las citas de los artículos quienes lo citan y permitiendo la autocita. Se calcula dividiendo el número de citas totales entre una frecuencia de tiempo de 2 años. Su fórmula es representada de la siguiente forma:

$$\text{FIA} = \# \text{ citas totales} / \text{frecuencia (2 años)}$$

4.2.2. Indicador de Impacto por Artículo de donde Proviene la Cita (IIAPC).

Este indicador ayudará a identificar si el artículo quien cita al artículo principal tiene mayor impacto, con ello conocer si el artículo principal está contribuyendo a una investigación con mayor desarrollo. Se calcula sumando el impacto del artículo principal más el impacto del artículo quien cita entre 2 en un lapso de 2 años. Su fórmula es representada de la siguiente forma:

$$\text{IIAPC} = \text{FIA (artículo principal)} + \text{FIA (artículo que citó)} / 2$$

4.2.3. Indicador de Impacto por Artículo Ponderado (IIAP).

Este indicador mostrará un impacto más justo de un artículo a diferencia del FIA, ya que es el promedio de la suma del impacto de cada artículo que cita al artículo principal. Este indicador se obtiene de la suma del impacto de cada artículo entre el

número total de citas en un periodo de 2 años. Su fórmula es representada de la siguiente forma:

$$\text{IIAP} = \text{FIA1} + \text{FIA2} + \text{FIAn} / \# \text{ Artículos}$$

4.2.4. Indicador de Impacto por Artículo en *Twitter* (IIAT).

Este indicador mostrará el impacto que tiene un artículo en la red social *Twitter*. Se calcula mediante el número de *tweets* sobre un periodo de tiempo de 6 meses. Su fórmula es representada de la siguiente forma:

$$\text{IIAT} = \# \text{ tweets} / \text{Frecuencia (6 meses)}$$

4.2.5. Indicador de Impacto por Artículo en *Facebook* (FIAF).

Este indicador mostrará el impacto que tienen los artículos en la red social *Facebook*. Se calcula mediante el número de *likes* que tiene un artículo en un periodo de tiempo de 6 meses. Su fórmula es representada de la siguiente forma:

$$\text{FIAF} = \# \text{ likes} / \text{Frecuencia (6 meses)}$$

Cabe señalar que en los indicadores para redes sociales se han elegido 6 meses por que el tiempo de vida de una publicación en ellas es más corto que en una fuente normal.

4.2.6. Indicador de Impacto por Artículo Sin Interacción Mutua (IIASIM).

Este indicador mostrará el impacto que tiene un artículo pero omitiendo la autocita, ya que reflejará el impacto sobre otros artículos de autores diferentes, es decir, así el investigador verá cual es el interés que provoca su artículo en otros investigadores. Se calcula por el número de citas sin auto citas entre un lapso de tiempo de 2 años. Su fórmula es representada de la siguiente forma:

$$\text{IIASIM} = \# \text{citas totales} / \text{frecuencia (2 años)}$$

4.3 Presentación

Para la presentación de los datos extraídos se propone una aplicación Web en la que los investigadores consulten el impacto de sus artículos, como también qué artículo tiene el mayor impacto en un año específico, cuál es el artículo más citado, el artículo de mayor impacto en una red social, conocer el impacto que tienen los artículos quienes citan al artículo principal y conocer información pertinente sobre las citas como son: autores, nombre del artículo, fecha de publicación, dónde lo publicaron y lugar, entre otros.

5. Desarrollo de la Investigación

En la figura 2 se muestra el proceso de una forma muy general para que la herramienta llegue a obtener el impacto de los artículos de forma automática desde la ontología de productividad hasta la aplicación Web para la visualización de resultados, así como también la tecnología que se usará para que cada proceso se lleve a cabo. Como primer caso se consulta, mediante el uso de Jena, el nombre de los artículos en la ontología de productividad del Instituto, para que posteriormente se envíen a la herramienta de extracción. Una vez que el extractor tiene la lista de artículos, buscará cada uno de ellos en las fuentes indicadas en la ontología de indicadores, al encontrarlo descargará tanto el número de citas con un periodo de 2 años, como también información relevante de cada una de las citas ya mencionado en la descripción de la capa de presentación.

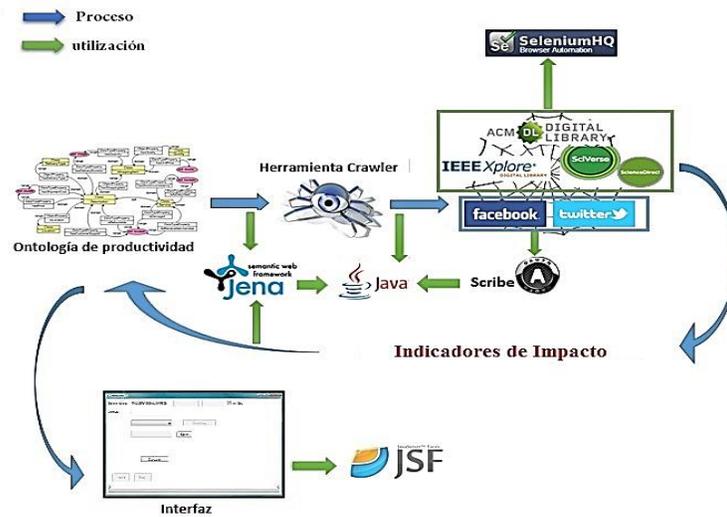


Figura 2. Ilustración del proceso de la herramienta para la obtención de impacto de los artículos de los investigadores del Instituto Tecnológico de Orizaba.

Como se mencionó anteriormente, para extraer los datos de las diferentes fuentes es necesario utilizar el marco de trabajo *Selenium*; para el caso de *Facebook* se extraerá el número de *likes* que tenga el artículo en el perfil del investigador o en la cuenta del programa de maestría en esta red; en el caso de *Twitter* también con las mismas cuentas, pero en dicha red, se extraerá el número de *tweets*. Una vez obtenido el número de citas e la información posteriormente se realiza el cálculo de cada uno de los indicadores mencionados en el apartado anterior; ya realizado el cálculo, con el auxilio de *Jena*, se guarda el resultado de los indicadores para cada artículo en la ontología de productividad y la información de las citas en la ontología de indicadores. Por

último, ya realizada la extracción y actualización de las ontologías, la aplicación Web visualizará el resultado de los indicadores para cada artículo como información a detalle de cada una de las citas que tengan.

6. Discusión

La limitación de las fuentes seleccionadas (ACM, IEEE, ScienceDirect) se debe a que proporcionan información detallada de las citas bajo acceso libre pese a ser bibliotecas virtuales de suscripción, también se les considera líderes en su tipo e incluyen a todas las revistas indizadas del área.

En el caso de redes sociales *Facebook* y *Twitter* se eligen por ser las que tienen el mayor impacto en la actualidad y se puede acceder a la información necesaria para el cálculo de indicadores mediante el protocolo OAUTH. La justificación de utilizar indicadores distintos a los usados actualmente en el medio es que, en programas de posgrado con orientación profesional, es menos frecuente la publicación en revistas indizadas, lo que da como resultado que se desconozca el impacto de muchos artículos, los cuales, es posible que sean de relevancia en el área académica aunque no se hayan publicado en una revista de alto impacto.

7. Resultados parciales

La primera versión del extractor de citas está formada por una serie de componentes que se muestran en la figura 3. El extractor ya obtiene la lista de artículos de la ontología de productividad y elige el Extractor específico a utilizar para extraer la información pertinente de cada fuente, pues dicha información no está representada de la misma manera en todas las páginas.

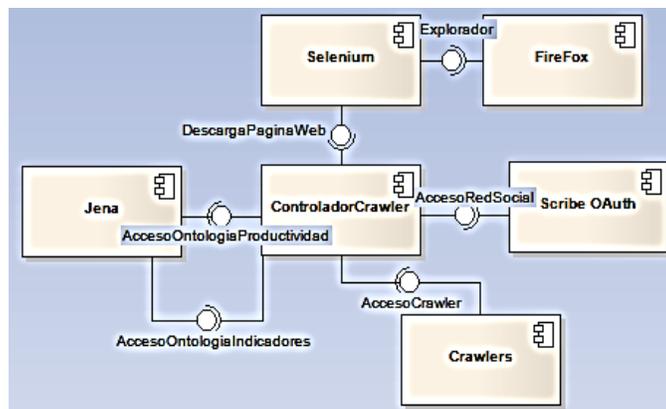


Figura 3. Diagrama de componentes del extractor de citas.

En la tabla 1 se muestra el resultado de una primera muestra de artículos de cuatro de los investigadores del Instituto Tecnológico de Orizaba: Giner Alor Hernández, Ulises Juárez Martínez, María Antonieta Abud Figueroa, Ana María Chávez Trejo, este resultado cuenta con el número de citas y el impacto de sólo 3 indicadores ya que falta trabajar con la extracción en redes sociales. También se muestra el resultado de un artículo externo “*Analysis of recommendation algorithms for e-commerce*” [28] para demostrar el resultado con un artículo que rebasa las 100 citas.

Tabla 1. Resultados parciales de los indicadores de impacto.

Investigadores	Artículo	Citas	FIA	IIAP	IIASIM
Giner Alor-Hernandez, Alberto Aguilar-Lasserre, Ulises Juarez-Martinez, Ruben Posada-Gomez, Guillermo Cortes-Robles, Mario Alberto Garcia-Martinez, Juan Miguel Gomez-Berbis, Alejandro Rodriguez-Gonzalez	HYDRA: A Middleware oriented Integrated Architecture for e-procurement in Supply Chains	1	0.0416	0.0416	0.0
Ulises Juarez-Martinez, Jose Oscar Olmedo-Aguirre	A Join Point Model for Fine Grained Aspects	1	0.0416	0.0	0.0
Giner Alor-Hernandez, Ruben Posada-Gomez, Ulises Juarez-Martinez, Gustavo Pelaez-Camarena, María Antonieta Abud-Figueroa	VEGITO: A Virtual Enterprise Generator	0	0.0	0.0	0.0
Giner Alor-Hernandez, Ulises Juarez-Martinez, Ruben Posada-Gomez, Ana Maria Chavez-Trejo, Jose Saul Rocha-Aragon	Defining an SOA for Stock Quote Management	0	0.0	0.0	0.0
Alor-Hernández G, Sánchez-Cervantes	ITOHealth: a multimodal	0	0.0	0.0	0.0

Investigadores	Artículo	Citas	FIA	IIAP	IIASIM
JL, Juárez-Martínez U , Posada-Gómez R, Cortes-Robles G, Aguilar-Lasserre A.	middleware-oriented integrated architecture for discovering medical entities				
Alejandro Rodríguez, Myriam Mencke, Giner Alor-Hernandez , Ruben Posada-Gomez, Juan Miguel Gomez, Alberto A. Aguilar-Lasserre	MEDBOLI: Medical Diagnosis Based on Ontologies and Logical Inference	2	0.0833	0.0208	0.0
Juan Miguel Gomez, Giner Alor-Hernandez , Ruben Posada-Gomez, Ma. Antonieta Abud-Figueroa , Angel Garcia-Crespo	SITIO: A Social Semantic Recommendation Platform	0	0.0	0.0	0.0
Giner Alor-Hernandez , Alberto Aguilar-Lasserre, Ulises Juarez-Martínez , Ruben Posada-Gomez, Guillermo Cortes-Robles, Mario Alberto Garcia, Martinez, Juan Miguel, Gomez, Myriam Mencke, Alejandro Rodriguez Gonzalez	A Hybrid Architecture for E Procurement	0	0.0	0.0	0.0
Ulises Juarez-Martínez , José Oscar Olmedo-Aguirre	Énfasis: a model for local variable crosscutting	1	0.0416	0.0416	0.0
Ulises Juarez-Martínez , Giner Alor-Hernandez	Advise Weaving in Énfasis	0	0.0	0.0	0.0

Investigadores	Artículo	Citas	FIA	IIAP	IIASIM
Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl	Analysis of recommenda- tion algo- rithms for e- commerce	147	6.125	1.547	0.0

Para mostrar la información extraída de las citas desde el extractor, en la figura 4 se muestra el resultado de la búsqueda individual del artículo “*HYDRA: a middleware-oriented integrated architecture for e-procurement in supply chains*” [29] del investigador Giner Alor Hernández, tiene 1 cita que corresponde al artículo “*Query expansion methods and performance evaluation for reusing linking open data of the European public procurement notices*” [30] que tiene como autor principal a José María Álvarez. También se muestra la bitácora del Controlador donde se observa que obtuvo el número de citas totales, el número de citas que están dentro de la diferencia de 2 años respecto al año actual, el número de auto citas y también los datos relevantes de cada cita como son autores, fecha de publicación, año, descripción, nombre del artículo, la URL de donde está localizado y cuántas citas tiene a su vez. Cabe señalar que la contabilidad de citas es realizada cuando el artículo es encontrado en alguna de las fuentes, si el artículo a buscar no es encontrado o no existe, simplemente el extractor no contabiliza las citas ni la información de ellas.

```

Artículo a buscar: HYDRA: a middleware-oriented integrated architecture for e-procurement in supply chains
Artículo encontrado en ACM link: http://dl.acm.org/citation.cfm?id=1880459.1880460&coll=DL4dl=GUIEH&CFID=2073655324&CFTOKEN=42560873
Autor principal: Giner Alor-Hernandez
numero de citas totales: 1
numero de citas con diferencia de 2 años al año actual: 1
numero de autocitas: 0
*****
Cita: Jose Maria Álvarez , José Emilio Labra , Ramón Calmeau , Ángel Marín , José Luis Marín , Query expansion methods and performance evaluation for reusing
ruta: http://dl.acm.org/citation.cfm?id=2075618&amp;CFID=207365548&amp;CFTOKEN=40929362
Autores:
Jose Maria Álvarez
José Emilio Labra
Ramón Calmeau
Ángel Marín
José Luis Marín
nombre de artículo: Query expansion methods and performance evaluation for reusing linking open data of the European public procurement notices
description: The aim of this paper is to present some methods to expand user queries and a performance evaluation to retrieve public procurement notices in t
fuente: http://dl.acm.org/
revista publicada: Proceedings of the 14th international conference on Advances in artificial intelligence: spanish association for artificial intelligence
fecha: November 07-11
año: 2011
lugar: La Laguna, Spain
Número de citas que tiene este artículo: 1
Nombre de los artículos de estas citas de este artículo:
Towards a theoretical foundation for the harmonization of linked data
Artículo No encontrado en ScienceDirect
Artículo No encontrado en IEEE
    
```

Figura 4. Resultado de extracción para el artículo “*HYDRA: a middleware-oriented integrated architecture for e-procurement in supply chains*”, para este resultado el artículo no se encontró en ScienceDirect ni IEEE

8. Conclusiones

La combinación de diferentes tecnologías y protocolos emergentes como la Minería de Contenido, las ontologías, las redes sociales y OAUTH entre otros, con técnicas de Ingeniería de Software sofisticadas como la arquitectura en capas y la creación de componentes, permite el desarrollo de aplicaciones complejas bajo aproximaciones iterativas. Estas aplicaciones permiten obtener de manera más fácil información relevante para los usuarios con miras a procesarlas para la adquisición de conocimiento específico, en este caso el impacto del trabajo de los investigadores. Si bien es cierto que ya existen en el mercado herramientas que obtienen datos parecidos, la presente propuesta cuenta con las ventajas de realizar el trabajo de forma automática, es decir, no es necesario indicar cada uno de los artículos sino que los toma de las fuentes propias de la Institución, la búsqueda de citas se realiza en fuentes confiables previamente definidas y, además, considera indicadores especiales, también parte de la propuesta, que toman en cuenta la situación particular del programa de posgrado y que se refleja a nivel de artículo y no de publicación.

9. Trabajo a futuro

Las siguientes etapas incluyen la creación de la capa responsable del cálculo de los indicadores así como de la aplicación Web que permita la consulta específica de los datos obtenidos de las citas.

También se tiene considerado un Análisis de Sentimientos, para definir si los comentarios publicados ahí, con respecto a los artículos, se pueden considerar como favorables o desfavorables a estos.

Agradecimientos

Este trabajo cuenta con apoyo por parte del Consejo Nacional de Ciencia y Tecnología (CONACyT).

Aclaración

Las referencias a ACM, IEEE, ScienceDirect no implican de manera alguna una vinculación directa de su parte con el presente trabajo.

Referencias

1. G. Buela Casal. Evaluación de la calidad de los artículos y de las revistas científicas: Propuesta del factor de impacto ponderado y de un índice de calidad. *Psicothema*, pp. 23-35, 2003.
2. N.PK Nigam A. Citation Index and Impact factor. *Indian J Dermatol Venereol Leprol*, p. 511, 2012. with Graphs (MLG '10), ACM, pp. 86-93, 2010.
3. U. Herb. OpenAccess Statistics: Alternative Impact Measures for Open Access Documents? An Examination How to Generate Interoperable Usage Information From Distributed Open Access Services. IFLA, pp. 165-178, 2012.
4. M. and Tumasjan Spörrle. Using Search Engine Count Estimates as Indicators of Academic Impact: A Web-based Replication of Haggbloom et al. *The Open Psychology Journal*, pp. 12-18, 2011.
5. G. Eysenbach. Can tweets predict citations? Metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. In *Journal of Medical Internet Research*, 2011.
6. M. Krauthammer Perry Evans. Exploring the use of social media to measure journal article impact. *Proceedings of the AMIA Annual Symposium*, pp. 374-381, 2011.
7. H. Piwowar. LSE - London School of Economics and Political Science. <http://blogs.lse.ac.uk/impactofsocialsciences/2012/04/04/31-flavours-research-impact/>, abril de 2012.
8. R. Baeza Yates. Tendencias en minería de datos de la Web. *El profesional de la información*, pp. 5-10, 2009.
9. P.M. Joshi and S. Liu. Web document text and images extraction using DOM analysis and natural language processing. *Proceedings of the 9th ACM symposium on Document engineering (DocEng '09)*, ACM, pp. 218-221, 2009.
10. H. Davulcu, A. Joglekar, A. More, S. Mukherjee, S. Patil, and I. V. Ramakrishnan. P. Choudhari. YellowPager: a tool for ontology-based mining of service directories from web sources. *ACM*, pp. 458-458, 2002.
11. P. Millet and J. Claude Heudin. Web Mining in the EVA Intelligent Agent Architecture. *IEEE Computer Society*, pp. 368-371, 2007.
12. F. Coetzee, E. Glover, G. Flake, D. Pennock, B. Krovetz, F. Nielsen, A. Kruger, and L. Giles Steve Lawrence. Persistence of information on the web: analyzing citations contained in research articles. *Proceedings of the ninth international conference on Information and knowledge management (CIKM '00)*, pp. 235-242, 2000.
13. T. Raiko, T. Tiinanen, and E. Hyv Katariina Nyberg. Document classification utilising ontologies and relations between documents. *Proceedings of the Eighth Workshop on Mining and Learning*.
14. R. Pedraza-Jiménez, L. Codina, and C. Rovira. Web semántica y ontologías en el procesamiento de la información documental. *El profesional de la información*, pp. 569-578., 2007.
15. Grupo de trabajo W3C. OWL Web Ontology Language Use Cases and Requirements. <http://www.w3.org/2007/09/OWL-Overview-es.html>.
16. Heaton, J. HTTP Programming Recipes for Java Bots. Heaton Research Inc. 2007.
17. Selenium. <http://docs.seleniumhq.org/>, abril de 2013.
18. Apache Jena (2013, Abril) Apache Jena [Online]. <http://jena.apache.org/>.
19. OAUTH (2013, Abril) OAUTH [Online]. <http://oauth.net/>.
20. Scribe Java. (2013, Abril) GitHub [Online]. <https://github.com/cdryden/scribe-java>.
21. ACM. (2013, Feb.) ACM. [Online]. <http://dl.acm.org/>.

22. IEEE Explore. <http://ieeexplore.ieee.org/Xplore/guesthome.jsp>. Febrero de 2013.
23. Science Direct. (2013, Feb.) Science Direct. [Online]. <http://www.sciencedirect.com/>.
24. Facebook. (2013, Feb.) Facebook. [Online]. <http://www.facebook.com/>.
25. Twitter. (2013, Feb) Twitter. [Online]. <https://twitter.com/>.
26. Source.net/open-source/crawlers.
27. Protege Project. (2013, Feb) Protege Project. [Online]. <http://protege.stanford.edu/>.
28. Java source. (2013, Feb) Java source. [Online]. <http://java-source.net/open-source/crawlers>.
29. Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2000. Analysis of recommendation algorithms for e-commerce. In Proceedings of the 2nd ACM conference on Electronic commerce (EC '00). ACM, New York, NY, USA, 158-167.
30. G. Alor-Hernandez., A. Aguilar-Lasserre, U. Juarez-Martinez, R. Posada-Gomez, G. Cortes-Robles, M. A. Garcia-Martinez & A. Rodriguez-Gonzalez. HYDRA: a middleware-oriented integrated architecture for e-procurement in supply chains. Transactions on computational collective intelligence I (pp. 1-20). 2010.
31. J. María Álvarez , J. Emilio Labra , R Calmeau , Á.l Marín , J. L. Marín. Query expansion methods and performance evaluation for reusing linking open data of the European public procurement notices. Proceedings of the 14th international conference on Advances in artificial intelligence: spanish association for artificial intelligence, November 07-11, 2011, La Laguna, Spain.

Análisis comparativo de marcos de trabajo para el desarrollo de servicios (Cloud Services) y aplicaciones en la nube (Cloud Computing)

Rafael Antonio Fernández Domínguez, Giner Alor Hernández

División de Estudios de Posgrado e Investigación
Instituto Tecnológico de Orizaba
rafafallo1@gmail.com, galor@itorizaba.edu.mx

Resumen. Hoy en día, las tecnologías para la Internet aumentan a pasos agigantados, provocando la existencia de un gran número de proveedores de servicios en la nube como: Google AppEngine, Zoho, Amazon E2C, entre otros; esto a su vez hace que las empresas adoptantes sean más exigentes al adquirir un servicio. En este sentido, este trabajo propone un análisis comparativo entre los diferentes marcos de trabajo para el desarrollo de servicios y aplicaciones en la nube, considerando aspectos de calidad como: multi-tenencia, confiabilidad, escalabilidad, costo, rendimiento e interfaces de programación de aplicaciones.

Palabras clave: Cloud computing, PaaS, SaaS, IaaS, quality attributes, frameworks.

1. Introducción

La computación en la nube (Cloud Computing) se define como un modelo que permite el acceso de red conveniente y bajo demanda, de un conjunto compartido de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que son rápidamente aprovisionados y liberados con un mínimo esfuerzo de gestión o interacción de servicios [1,2]. Es una tecnología orientada a la Web, y surge para combinar ciertos aspectos para la mejora en el almacenamiento y ventajas de las empresas, como la reducción de costos, de producción y de mantenimiento. Se confunde fácilmente con el Grid Computing que surge a mediados de 1990, se deriva de la red de energía eléctrica para destacar sus características como la capacidad de penetración, sencillez y fiabilidad. La demanda a gran escala de aplicaciones científicas requieren más poder de cómputo que un grupo dentro de un mismo dominio (por ejemplo, un instituto) podría proporcionar; y debido a la interconexión rápida por Internet, institutos científicos fueron capaces de compartir y agregar recursos geográficamente distribuidos, incluidos los sistemas de racimo, las instalaciones de almacenamiento de datos y fuentes de datos pertenecientes a diferentes organizaciones [3]. Grandes empresas de Software forman parte de esta tecnología brindando

diversos servicios a los usuarios, estos servicios se clasifican en tres tipos diferentes: PAAS, SAAS e IAAS [1, 2].

Plataforma como servicio (PaaS): Ofrece un entorno de desarrollo como un servicio, donde las aplicaciones se desarrollan utilizando un conjunto de lenguajes de programación y herramientas. Estos servicios incluyen almacenamiento de desarrollo, integración, pruebas o recursos para completar el ciclo de vida de servicios [1]. El consumidor no gestiona ni controla la infraestructura de nube subyacente, incluyendo la red, servidores, sistemas operativos, o el almacenamiento, pero tiene el control sobre las aplicaciones implementadas y posiblemente de alojamiento de aplicaciones configuraciones de entorno [2]. Por otra parte, el Software como servicio (SaaS) es el uso de las aplicaciones del proveedor que se ejecutan en una nube de infraestructura. Las aplicaciones son accesibles desde diferentes dispositivos cliente a través de una interfaz de cliente ligero como un browser. El consumidor no gestiona ni controla la infraestructura de nube subyacente, servidores, sistemas operativos, almacenamiento o incluso capacidades de aplicaciones individuales [2]. Finalmente, la Infraestructura como servicio (IaaS) permite gestionar un gran conjunto de recursos de computación, tales como el almacenamiento y la capacidad de procesamiento y ejecución de aplicaciones dentro de las máquinas virtuales, donde el consumidor es capaz de instalar y ejecutar software arbitrario, que incluye sistemas operativos y aplicaciones [1]. El consumidor no gestiona ni controla la infraestructura de nube subyacente pero tiene el control de los sistemas operativos, almacenamiento, aplicaciones desplegadas, y el control posiblemente limitado de componentes de red seleccionados (por ejemplo, firewalls de host) [2].

Desafortunadamente para los usuarios, día a día aumenta la existencia de una gran cantidad de marcos de trabajo para desarrollar aplicaciones y servicios en la nube. Esto conlleva al aumento en la exigencia de los usuarios al seleccionar el marco deseado y buscan la mejor opción basándose en el uso de atributos que demuestren la calidad de cada marco y así dar seguridad a su selección. Sin embargo, para esto se requiere de un arduo y agotador trabajo ya que no existe material que auxilie o proporcione apoyo y soporte para hacer este trabajo de una manera más rápida y sencilla. Al respecto, existen algunos análisis comparativos orientados a este tema [1, 3], sin embargo, son escasos de información referente a los diversos marcos de trabajo basados en aspectos de calidad. Por lo tanto, este trabajo se enfoca en el análisis comparativo de marcos de trabajo basado en aspectos de calidad para el desarrollo de aplicaciones y servicios en la nube aportando la facilidad en la selección de un marco de trabajo para los usuarios que así lo requieran y que se adapte a las necesidades del mismo.

2. Trabajos Relacionados

En años recientes se han propuesto diversos trabajos enfocados a la computación en la nube, a continuación se presentan algunos de ellos.

En [4] se presenta un estudio de diversos servicios en la nube. Se identifican y explican sus principales características, posteriormente se organizan estas características y se propone una taxonomía y estructura de árbol basándose en taxonomías y en tablas de comparación existentes. Esta taxonomía permite clasificaciones rápidas de los diferentes servicios de Computación en la Nube y hace que sea más fácil su comparación dando solución a la falta de estandarización. En [5] se plantea el desarrollo de un marco de trabajo llamado (SMI) Service Measure Index, que consiste en la aplicación de medidas coherentes y significativas que se diseñaron para permitir la comparación de los actuales servicios basados en la nube o proveedores de servicios disponibles. Este marco de trabajo se propuso por el Cloud Services Measurement Initiative Consortium (CSMIC) el cual nace para hacer frente a la necesidad de que toda la industria a nivel mundial tenga la posibilidad de calcular los beneficios y riesgos de los servicios de computación en la nube. En [6] se presenta de manera general el concepto, la historia, las ventajas y desventajas de la computación en la nube, así como el esfuerzo para crear una normalización de la misma. La computación en nube se introdujo por primera vez por Eric Schmidt en su intervención en las conferencias Search Engine Strategies en 2006 y existen muchas definiciones y metáforas de la computación en la nube. Según el autor de este artículo, la computación en la nube es un tipo de técnica de computación donde los servicios de TI (Tecnologías de la Información) son provistos por grandes unidades de computadoras de bajo costo conectados en red.

En [7] se examinan algunas perspectivas de Computación en la nube así como algunas características esenciales de la misma, basadas en definiciones dadas por el NIST (National Institute of Standard and Technology) como lo son los SPI (Software as a Service, Platform as a Service e Infrastructure as a Service) que son los tres modelos de servicio en la nube, los 4 modelos de prestación de la nube (Pública, Privada, Híbrida y de Comunidad) y la relación que tiene un cliente con dichos modelos. En [8] se propone un marco de trabajo y un mecanismo que mida la calidad y dé prioridad a servicios en la nube, ya que debido a la gran diversidad de servicios en la nube disponibles es difícil decidir qué servicio utilizar y en qué se basa para su selección, puesto que hoy en día no existe algún marco de trabajo que permita evaluar dichos servicios. Por lo tanto, se presenta el primer marco de trabajo SMICloud, que permite medir de forma sistemática todos los atributos de QoS (Quality of Service) propuestos por CSMIC y clasificar los servicios en la nube. En [9] se propone una arquitectura para la asignación de recursos orientada a la alta calidad del servicio y el cumplimiento del Service Level Agreements (SLA), para ello se definen los requisitos de arquitectura en la nube asegurando la calidad de servicio basado en las métricas definidas por CSMIC, quien define algunos Service Measure Index (SMI). En [10] se proporciona información a detalle de criterios de seguridad que los próximos clientes de los servicios en la nube considerarán para seleccionar de manera eficiente una nube pública o un Cloud Service Provider (CSP) que satisfaga sus necesidades, ya que para la selección de una nube se requiere considerar los riesgos potenciales que puedan afectar la integridad de la misma afectando al cliente. Algunos de los criterios mencionados en el artículo están basados en la norma ISO 27000 e ISO 27002, de igual manera se mencionan certificaciones que son utilizados por los CSPs en base a la seguridad. En [11] se plantea el objetivo de dar una mejor comprensión del tema identificando y clasificando las principales preocupaciones y soluciones de seguridad en la

nube, de igual manera, se propone una taxonomía dando una visión general de lo que es la seguridad actualmente en la computación en la nube. Debido al creciente interés en el tema de la seguridad, hay un esfuerzo constante para evaluar las tendencias actuales de seguridad para la Computación en la nube. Algunas organizaciones que se identifican para abarcar este tema son: ENISA, CSA y NIST que se denominan marcos de seguridad.

En [12] se revisan los desafíos relacionados con los conceptos de confianza y gestión de Service Level Agreements (SLA) y se analizan los marcos existentes de los acuerdos de nivel de servicio en diferentes ámbitos, como los servicios Web y la red. Se ponen de relieve muchas cuestiones, tanto en el desarrollo de los Service Level Agreements (SLA) y los modelos de rendimiento para computación en nube, que constituyen los ricos campos de investigación futura. En [13] se propone un marco de trabajo de comparación que contiene las características de Computación en la nube. Este marco de trabajo se aplica en cuatro modelos de procesos establecidos para la ingeniería de requerimientos que son RUP, XP, Volere y VModel, con el fin de estudiar a qué grado se aplican a los requisitos específicos de las soluciones basadas en la nube. El marco de comparación se desarrolló sobre 16 características en cuatro categorías, y representa una oportunidad para base de un estudio de la literatura disponible.

En [14] se presenta una selección de temas de Computación en la nube centrándose más en cuestión de seguridad ya que es muy importante tener en cuenta la seguridad y privacidad a la hora de diseñar y utilizar servicios en la nube. En [15] se presenta un estudio exhaustivo acerca de los mecanismos de elasticidad disponibles en la actualidad, con el objetivo de proporcionar una mejor comprensión de los temas actuales de investigación en este campo, mediante la realización de una encuesta la cual se divide en tres partes. En la primera, se presenta una clasificación de las soluciones de elasticidad; en la segunda se describen y clasifican las soluciones desarrolladas y por último, algunas cuestiones pendientes y desafíos relacionados con el uso de la función de elasticidad en la computación en nube.

3. Atributos de Calidad Aplicables a Servicios en la Nube

Los atributos de calidad son aquellos puntos que brindan a los usuarios la confianza al seleccionar un marco de trabajo para el desarrollo en la nube. En la tabla 1 se presentan los criterios de calidad que presentó el “Cloud Service Measurement Initiative Consortium” (CSMIC) en su marco de trabajo SMI para medir la calidad de los marcos de trabajo en la nube [16].

Tabla 1. Tabla de Criterios de Calidad

CATEGORIA	CRITERIO
Responsabilidad ¿Contamos también con la organización de proveedor?	<ul style="list-style-type: none"> ➤ Conformidad ➤ Contratación de experiencia ➤ Propiedad de los datos ➤ Facilidad para hacer negocios ➤ Gobernanza ➤ Propiedad ➤ Proveedor de la estabilidad del negocio ➤ Certificaciones de proveedores ➤ Contrato de prestación/verificación de SLA ➤ Proveedor de ética ➤ Requisitos personal del proveedor ➤ Proveedor de la cadena de suministro ➤ Capacidades de seguridad
Agilidad ¿Es fácil ser cambiado y qué tan rápido?	<ul style="list-style-type: none"> ➤ Adaptabilidad ➤ Capacidad ➤ Elasticidad ➤ Extensibilidad ➤ Flexibilidad ➤ Portabilidad ➤ Escalabilidad
Garantía ¿Qué tan probable es que el servicio funcionará como se esperaba?	<ul style="list-style-type: none"> ➤ Disponibilidad ➤ Datos geográficos / políticos ➤ Mantenimiento ➤ Recuperación ➤ Confiabilidad ➤ Resistencia / tolerancia a fallos ➤ Servicio de estabilidad ➤ Utilidad
Financiero ¿Cuánto es?	<ul style="list-style-type: none"> ➤ Adquisición y la costa transición ➤ Costo en curso ➤ Beneficios o costos compartidos
Rendimiento ¿Hace lo que necesitamos?	<ul style="list-style-type: none"> ➤ Precisión ➤ Funcionalidad ➤ Interoperabilidad ➤ Servicio de tiempo de respuesta
Seguridad y privacidad ¿Es el servicio seguro y protege la privacidad?	<ul style="list-style-type: none"> ➤ Control de acceso y gestión de privilegios ➤ Datos geográficos / políticos ➤ Integridad de los datos ➤ Privacidad de los datos y la pérdida de Datos ➤ Seguridad física y ambiental ➤ Proactiva de amenazas y gestión

CATEGORIA	CRITERIO
	de vulnerabilidades ➤ Retención / disposición
Usabilidad ¿Es fácil de prender y de usar?	➤ Accesibilidad ➤ Requerimientos personales del cliente ➤ Capacidad de instalación ➤ Facilidad de aprendizaje ➤ Operatividad ➤ Idoneidad ➤ Transparencia ➤ Comprensibilidad

Al identificar los atributos de calidad, se llevó a cabo la selección de los más relevantes, los cuales se muestran a continuación:

1. Adaptabilidad
2. Capacidad
3. Elasticidad
4. Extensibilidad
5. Flexibilidad
6. Portabilidad
7. Escalabilidad
8. Accesibilidad

Estos elementos se seleccionaron porque son los atributos que brindan la confianza al usuario en cuanto a la agilidad y la usabilidad del servicio en la nube requerido, y forman parte de la misma clasificación.

4. Marcos de trabajo para el desarrollo de servicios y aplicaciones para la computación en la nube PAAS

Los marcos de trabajo orientados al desarrollo de aplicaciones para la nube permiten la creación de aplicaciones capaces de desplegarse en las mismas. Existe gran variedad de marcos de trabajo, a continuación se presentan algunos de los más importantes con una breve descripción:

- **Google App Engine:** Es una oferta PaaS, que proporciona soporte para Python y Java. AppEngine permite crear aplicaciones Web escalables, sin necesidad de hardware complejo subyacente y capas de software. Google abstrae esas capas y permite concentrarse plenamente en la aplicación [17].

- **Aneka:** Es una red basada en la plataforma-como-un-Servicio de desarrollo de aplicaciones (PaaS), que ofrece un entorno de ejecución y un conjunto de APIs que permiten a los desarrolladores crear aplicaciones personalizadas, mediante el uso de múltiples modelos de programación como la programación de tareas, Thread y programación MapReduce [18].
- **Force:** Es una plataforma de computación en la nube empresarial ofrecido por Salesforce. Permite a los vendedores de servicios el desarrollo y distribución de aplicaciones estables, seguras y escalables. Dos tecnologías clave de Force.com son multi-tenencia y metadatos [19].
- **Microsoft Azure:** La plataforma de servicios Azure es la oferta PaaS de Microsoft. Azure tiene una API específica para almacenar y recuperar datos llamados Services de SQL. El sistema subyacente para estos servicios SQL es Microsoft SQL Server. Aunque no todas las funcionalidades de Microsoft SQL Server están expuestas a través de la API, el usuario ejecuta las operaciones y el uso de un lenguaje de consulta SQL restringido [20].
- **Heroku:** Es una muy poderosa plataforma basada en Web que proporciona soporte para Ruby, JavaScript y Java Web. Se despliega directamente en un entorno PaaS, permite la migración rápida desde el desarrollo a la etapa de implementación. También contiene un editor de colaboración en tiempo real para su uso con un máximo de cinco personas [21].
- **Amazon Elastic MapReduce:** Es un servicio Web de Amazon (AWS), que utiliza Hadoop para ofrecer una funcionalidad MapReduce. En el fondo sí se basa principalmente en otros dos servicios AWS: Elastic Compute Cloud (EC2) y Simple Storage Service (S3) [22].
- **CloudFoundry:** Es una plataforma de código abierto como un servicio. Entorno ofrecido por VMware, proporciona el entorno para albergar múltiples lenguajes y marcos en una pila abierta de aplicaciones de software que se ejecutan tanto fuera como dentro del firewall. Las principales características de CloudFoundry son: Elección de los marcos de desarrollo, elección de servicios de infraestructura de aplicaciones, y la elección de las nubes [23].
- **OrangeScape:** OrangeScape permite construir completas aplicaciones de negocio utilizando una interfaz de estilo de modelado visual en un entorno 5GL. Esto se hace a través de Studio OrangeScape directamente en el browser de preferencia. Las aplicaciones desarrolladas con la plataforma OrangeScape proporcionan las mejores capacidades de escalabilidad, administración y control de los gastos que se adapte a cualquier proveedor como Google App Engine, Azure de Microsoft, IBM Smart Cloud, y Amazon EC2 [24].
- **OpenShift:** Es la plataforma de Red Hat Cloud Computing as a Service (PaaS) que ofrecen. OpenShift es una plataforma de aplicaciones en la nube, donde los desarrolladores de aplicaciones y los equipos tienen la opción de crear, probar, implementar y ejecutar sus aplicaciones. OpenShift se encarga de toda la infraestructura, middleware, y la gestión y permite a los desarrolladores centrarse en lo que mejor saben hacer: el diseño y la codificación de aplicaciones [25].
- **Mendix:** La Plataforma Mendix App permite a las empresas crear rápidamente aplicaciones Web y móviles, integrar a la perfección, desplegar al instante, gestionar de forma centralizada, y mantener sin esfuerzo sobre una base continúa. Las empresas utilizan la Plataforma Mendix para crear aplicaciones, implementar e in-

tegrar aplicaciones Web y empresariales móviles más rápido y mejor que nunca. [26].

- **Engine Yard:** Engine Yard es la plataforma líder como servicio (PaaS) para una amplia gama de aplicaciones en Ruby on Rails, PHP y Node.js. Tiene la flexibilidad y la transparencia de una plataforma altamente configurable y abierta. Una variedad de opciones de personalización están disponibles para la configuración, despliegue y gestión de su entorno de aplicación [27].
- **Oracle Cloud:** Oracle Cloud ofrece un amplio portafolio de software como un servicio de aplicaciones, la plataforma como servicio, y capacidades sociales, todo en una base de suscripción. Oracle Cloud ofrece un valor inmediato y la productividad de los usuarios finales, administradores y desarrolladores por igual a través de ricas e integradas funcionalidades [28].

En la tabla 2 se muestra una comparación de los marcos de trabajo para el desarrollo de aplicaciones y servicios en la nube mencionados anteriormente, presentando los diferentes lenguajes de programación que soporta cada uno y los IDE's que ofrecen un entorno de desarrollo para los mismos.

Tabla 2. Tabla Comparativa de los Marcos de Trabajo

MARCOS DE TRABAJO	LENGUAJES DE PROGRAMACION	IDE's
Google App Engine	Python, Java	Eclipse
Aneka	.NET	Aneka Cloud Management Studio
Force	Apex	Eclipse
Microsoft Azure	.NET, PHP, Java	Eclipse Visual Studio
Heroku	Ruby, Java, Python, Node.js	Toolbelt heroku
Amazon EC2	Java, Ruby,Perl, Python, PHP, C++	Eclipse Visual Studio
CloudFoundry	Java, Ruby, Scala, Python, PHP, Node.js	Spring Tool Suite (STS) Eclipse
OrangeScape	Python, java, JQuery, JSTL, Spring, hibernate, bigtable, c#, entre otras.	OrangeScape Studio
OpenShift	Java, Ruby, PHP, Perl, Python, Node.js.	Eclipse, JBoss Developer Studio
Mendix	Java.	Eclipse
Engine Yard	Ruby, PHP, Node.JS	...
Oracle Cloud	Java	Eclipse Oracle Developer NetBeans IDE

Después de analizar los diferentes marcos de trabajo para el desarrollo de aplicaciones en la nube se consideraron 8 de ellos que son los más importantes, ya que contienen mayor soporte, documentación y renombre y se mencionan a continuación:

1. Google App Engine
2. Force
3. Microsoft Azure
4. Amazon EC2
5. CloudFoundry
6. Oracle Cloud
7. Engine Yard
8. Mendix

5. Resultados del Análisis Comparativo

En esta sección se presentan los resultados de las pruebas parcialmente realizadas a los diferentes marcos de trabajo para el desarrollo de aplicaciones y servicios en la nube. Estas pruebas consisten en construir pequeñas aplicaciones que demuestren la accesibilidad, ya que este atributo asegura que el marco de trabajo puede ser accedido, comprendido, aprendido y utilizado por el usuario, de tal modo que se llevó a cabo el desarrollo y el despliegue en cada nube de una pequeña aplicación dando lugar al resultado que determinará si este atributo de calidad es aplicable a dicho marco.

La aplicación realizada es una simple aplicación que tiene lugar siempre al comienzo en la vida del desarrollador de software, un simple y sencillo “Hola mundo” en cada plataforma.

Cabe mencionar que en cada marco de trabajo, la aplicación desarrollada se adaptó a la estructura y lenguaje requerido por los mismos utilizando los componentes que brindan con el fin de realizar una evaluación completa.

Un factor importante en el análisis de la accesibilidad es la verificación de licencias en cada marco de trabajo ya que esto puede ser un impedimento para ciertos usuarios. En la tabla 3, se muestra un análisis de los marcos de trabajo que presentan una licencia que es de pago para su uso y una licencia que es una versión gratuita.

Tabla 3. Tabla Comparativa de Licencias en los Marcos de Trabajo

MARCOS DE TRABAJO	LICENCIA DE PAGO	VERSIÓN GRATUITA
Google App Engine	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Force	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Microsoft Azure	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Amazon EC2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
CloudFoundry	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Engine Yard	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Mendix	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Oracle Cloud	<input checked="" type="checkbox"/>	

Es importante mencionar que las Licencias tanto de pago como las gratuitas varían en sus características y costos.

Otros factores importantes en el análisis de la accesibilidad es la forma de navegación en cada plataforma ya que cada una posee una arquitectura diferente para gestionar las aplicaciones generadas, de la misma manera el acceso a sus herramientas para desarrollar, depurar y desplegar las aplicaciones en la nube.

Los resultados obtenidos en cada prueba realizada con respecto a la accesibilidad tomando en consideración los factores importantes mencionados anteriormente, se ven reflejados en la tabla 4. En esta tabla se puede observar en qué nivel de accesibilidad se considera cada marco de trabajo, dichos niveles se encuentran clasificados como:

- 1) “Alto” el cual significa que cuenta con sencillez en el proceso de navegación, fácil acceso a las herramientas para desarrollo, depuración y despliegue, documentación abundante para comprender y aprender cómo funciona dicho marco, y permite el acceso a ella mediante una licencia gratuita delimitada por tiempo o por capacidad.
- 2) “Medio” el cual es el nivel que contempla una navegación sencilla, se pueden obtener las herramientas proporcionadas fácilmente pero no cuenta con la documentación apta para principiantes, lo que provoca la dificultad para acceder a ella y brindar un buen servicio al desarrollar aplicaciones; además, la licencia gratuita proporcionada es demasiado limitada.

3) “Bajo”, en este nivel se encuentran los marcos de trabajo más complejos ya que su navegación es agotadora, las herramientas que proporciona son demasiado complejas para utilizarlas si no se tiene capacitación previa, la documentación proporcionada es compleja para su comprensión y la licencia proporcionada es limitada totalmente, o en algunos casos nula, solo admitiendo pagos por ella. Esto hace complejo el aprendizaje para desarrollo de aplicaciones en esta plataforma.

Tabla 4. Tabla de Resultados con base en la Accesibilidad.

NIVEL DE ACCESIBILIDAD			
MARCOS DE TRABAJO	BAJO	MEDIO	ALTO
URL de Prueba.			
Google App Engine Maestriaensistemascomp.Appspot.com			
Force No aplica			
Microsoft Azure No aplica			
Amazon EC2 http://miprimerappenv-exwa4zmepk.elasticbeanstalk.com/			
CloudFoundry miappcloudfoundry1.cloudfoundry.com			
Engine yard http://holamundoito.elasticbeanstalk.com			
Mendix https://rafaelfernandez100.mendixcloud.com/			
Oracle Cloud No aplica			

Después del análisis realizado se observa que la accesibilidad se encuentra presente en todas las plataformas en la nube, pero por desgracia para los usuarios varía en cada una de ellas haciéndose muy complejo su manejo.

6. Trabajo a Futuro

Como trabajo a futuro se considera la inclusión de marcos de trabajo adicionales como Aneka, Open Shift, heroku y orangeScape para realizar un análisis más completo. Así también, la inclusión en este análisis de nuevos atributos de calidad como lo son: Capacidad, Elasticidad, Extensibilidad, Flexibilidad, Portabilidad y Escalabilidad, desarrollando y desplegando en los diferentes marcos pequeñas aplicaciones, las cuales permitan obtener resultados que nos muestren a qué grado estos atributos de calidad son aplicados en cada plataforma en la nube.

7. Conclusiones

La computación en la nube es una tecnología que ha mostrado gran crecimiento con el paso de los años, y a la cual muchas empresas se han incorporado para prestar servicios. Esto incita un gran interés a las personas y las empresas por su uso pero es compleja la selección del servicio al desear adoptarla, ya que existen en la actualidad gran cantidad de servicios PAAS, como lo son Google App Engine, Amazon, Windows Azure, por mencionar algunas, cada uno ofreciendo sus ventajas y desventajas.

El análisis realizado a los trabajos relacionados permite ratificar la relevancia de la propuesta planteada en este trabajo. A pesar de que hay contribuciones al respecto de los marcos de trabajo para el desarrollo de aplicaciones para la nube, esta propuesta aborda aspectos que por lo menos aún no se reportan en otros trabajos. De igual manera, el análisis realizado a los trabajos relacionados revela las características que ofrecen los diferentes marcos de trabajo para el desarrollo de aplicaciones en la nube y de la misma manera se revelan los principales atributos que reflejan la calidad del software.

En este trabajo, el análisis realizado parcialmente a los marcos de trabajo, refleja la complejidad de algunos de ellos, así como la facilidad de otros para su manipulación y acceso, demostrando que el atributo de calidad “Accesibilidad” en todos los marcos es existente y tiene una función primordial pero con diferente grado de dificultad. Así también, se presenta la capacidad de tener acceso al marco de trabajo, la cantidad de información brindada, la facilidad de aprendizaje y la facilidad en el manejo de los componentes brindados, como lo son los plugins con los cuales se permiten desarrollar las aplicaciones.

Agradecimientos

Este trabajo fue patrocinado por el Consejo Nacional de Ciencia y Tecnología (CONACYT)

Referencias

1. María del Pilar Salas, Luis Omar Colombo. Cloud Computing: A Review of PaaS, IaaS, SaaS Services and Providers. Lámpasakos No. 7, Medellín, Colombia. 2012 Pp. 47-57.
2. Mark Lee Badger, Timothy Grance, Robert Patt-Corner, Jeffery M, Voas. Cloud Computing Synopsis and Recommendations. Technical Report. NIST Gaithersburg, MD, United States, 2012, pp 0-81.

3. Christof Weinhardt, Arun Anandasivam, Benjamin Blau, et al. Cloud Computing – A Classification, Business Models, and Research Directions. Business & Information Systems Engineering volumen 5, 2009. Karlsruhe, Germany. pp 392-399.
4. C.N. Höfer, G. Karagiannis. Cloud computing services: taxonomy and comparison. Processing of the GLOBECOM Workshops (GC Wkshps), 2010 IEEE, Springer. Enschede, the Netherlands. pp 81-94.
5. Jane Sigel, Jeff Perdue. Cloud Services Measures For Global Use, The Service Measure Index (SMI). Proceedings of the Service Research and Innovation Institute Global Conference, IEEE. Mountain View, CA, USA, 2012 pp. 411-415.
6. Ling Qian, Zhiguo Luo, Yujian Du, Leitao Guo. Cloud Computing: An Overview. Proceeding of the Cloud Computing, First International Conference, CloudCom 2009, Springer-Verlag Berlin Heidelberg. Beijing, China. pp. 626-631.
7. Vic (J.R.) Winkler. Chapter 2- Cloud Computing Architecture. Securing the Cloud, Cloud Computer Security Techniques and Tactics 2011, Elsevier. Waltham, Massachusetts, USA, pp 29–53.
8. Saurabh Kumar, Steve Versteeg , Rajkumar Buyya. A framework for ranking of cloud computing services. Future Generation Computer Systems 2012 Volume 29, Elsevier. Melbourne, Australia. pp 1012–1023.
9. Kouessi Arafat Sagbo, Pélagie Houngue. Quality architecture for resource allocation in cloud computing. Proceedings of the First European conference on Service-Oriented and Cloud Computing (ESOCC'12).
10. Vic (J.R.) Winkler. Chapter 8 – Security Criteria: Selecting an External Cloud Provider. Securing the Cloud, Cloud Computer Security Techniques and Tactics 2011, Elsevier. Waltham, Massachusetts, USA. pp 211–232.
11. Nelson Gonzalez, Charles Miers, Fernando Redigolo, Marcos Simplicio, Tereza Carvalho, Mats Naslund, Makan Pourzandi. A quantitative analysis of current security concerns and solutions for cloud computing. Proceedings of the IEEE Third International Conference on Cloud Computing Technology and Science CLOUDCOM '11. Washington, DC, USA, Pp 231-238.
12. Mohammed Alhamad, Tharam Dillon, Elizabeth Chang. A Survey on SLA and Performance Measurement in Cloud Computing. Processing of the On the Move to Meaningful Internet System (OTM2011). Springer, Australia, pp. 469–477.
13. Stefan Wind, Schrödl. Requirements Engineering for Cloud Computing: A Comparison Framework. Proceedings of Workshop on Information Systems Economics (WISE2010), Springer-Verlag Berlin Heidelberg. Augsburg, Germany, pp. 404-415.
14. Ainul Azila Che Fauzi, A. Noraziah, Tutut Herawan, Noriyani Mohd. Zin. On cloud computing security issues. In Proceedings of the 4th Asian conference on Intelligent Information and Database Systems (ACIIDS'12), Springer-Verlag, Berlin, Heidelberg, 2012. pp 560-569.
15. Guilherme Galante, Luis Carlos E. de Bona. A Survey on Cloud Computing Elasticity. Proceedings of the IEEE/ACM Fifth International Conference on Utility and Cloud Computing (UCC '12). IEEE Computer Society, Washington, DC, USA, 2012, pp. 263-270.
16. Jane Sigel, Jeff Perdue. A Cloud Services Measures for Global Use, The Service Measure Index (SMI). Proceedings of the Service Research and Innovation Institute Global Conference, IEEE. Mountain View, CA, USA, 2012, pp. 411-415.
17. Google team. 26-mar-2012. Google App Engine. 23-Feb-2013, <https://developers.google.com/appengine/?hl=es>.
18. Manjrasoft Pty Ltd. 18-Jun-2012 http://www.manjrasoft.com/aneka_architecture.html

19. Salesforce.com team. 23-Feb-2013, Force.com Cloud Computing Platform. 23-Feb-2013, <http://www.salesforce.com/platform/cloud-platform/>.
20. Microsoft team. 23-Feb-2013. Windows Azure: Plataforma en la nube de Microsoft. <http://www.windowsazure.com/es-es/>.
21. Heroku team. 23-Feb-2013. Heroku Cloud Application Platform. <http://www.heroku.com/>.
22. Amazon Web Services team. 23-Dic-2011 Amazon Elastic MapReduce (Amazon EMR). <http://aws.amazon.com/elasticmapreduce/>.
23. VMware team. 24-Ene-2013. Cloud Foundry. <http://www.cloudfoundry.com/>.
24. OrangeScape team. 23-Feb-2013. OrangeScape. <http://www.orangescape.com/paas/platform-as-a-service/>.
25. OpenShift team. 23-Mar-2012. OpenShift. <https://www.openshift.com/paas>
26. Mendix team. Mendix. 23-Feb-2013. <http://www.mendix.com/application-platform-as-a-service/>.
27. EngineYard team. 23-Feb-2013. EngineYard. <https://www.Engineyard.com/products/cloud>.
28. Oracle team. 23-Feb-2013. Oracle Cloud. https://cloud.oracle.com/mycloud/f?p=service:developer_service_features.

Desarrollo de una herramienta Web para monitorización y control de proyectos de software para empresas con nivel 5 de CMMI-DEV

D. A. Ramos Calvario¹, A. M. Chávez Trejo¹, Z.B. Osorio Martínez²

¹División de Estudios de Posgrado e Investigación Instituto Tecnológico de Orizaba,
²Intellego S.A. de C.V.
darc_b_30@hotmail.com, achavez@itorizaba.edu.mx,
elpkbj@gmail.com

Resumen. En este trabajo se exponen los avances en el desarrollo de la herramienta Web para la monitorización y control de los proyectos de software en empresas de alto nivel de madurez de CMMI-DEV, las cuales están invirtiendo tiempo y recursos financieros en la búsqueda de la mejora continua y en la calidad de sus procesos, desarrollando herramientas que ayuden a su monitoreo. El modelo CMMI-DEV cuenta con las áreas de "Monitorización y Control" así como "Desempeño de los Procesos Organizacionales", las cuales apoyan a la monitorización de las líneas base de desempeño de la empresa en sus subprocesos críticos, así como sus límites de control y confianza, a fin de tomar acciones correctivas apropiadas cuando el rendimiento se desvía significativamente. La herramienta que se propone en este trabajo, atenderá la inexistencia de herramientas documentadas que apoyen a las áreas antes mencionadas para monitorizar las líneas base en una empresa de alta madurez, esta herramienta será desarrollada para la empresa Intellego S.A. de C.V. la cual posee un nivel 5 de madurez [1]. La automatización pretende una reducción de hasta un 50% de esfuerzo para la recolección y análisis de información, adaptándola a la herramienta de producción diaria (Jira).

Palabras clave: Herramienta Web, Proceso de Desarrollo de Software, Modelo de Calidad, Monitorización y Control de Proyectos.

1. Introducción

Actualmente las empresas de desarrollo de software que desempeñan procesos de producción y que se encuentran evaluadas en modelos de calidad, buscan estar siempre a la vanguardia, por lo que están invirtiendo en automatizar sus procesos [2]. Las empresas relacionadas en la mejora continua, son aquellas con alto nivel de madurez en modelos de calidad como CMMI-DEV, el cual es un modelo de mejora de procesos para el desarrollo de software. Una alternativa para la mejora es la automatización de los procesos y subprocesos en el desarrollo de sus productos [3]. CMMI-DEV

establece 22 áreas clave de proceso en el nivel más alto de madurez [4], empresas con nivel 5 en este modelo, requieren monitorizar el comportamiento de sus subprocesos, es por ello que se requiere una comprensión del rendimiento de los proyectos en los sub procesos que afectan de forma significativa los objetivos del proyecto y/o su organización, de modo que se identifiquen causas profundas que originan las desviaciones y posterior a ellos se definan mejoras que estabilicen el comportamiento del proyecto. El proceso a aplicar contempla actividades de verificación y validación mediante pruebas formales y de aceptación dentro del entorno real de trabajo, las cuales serán realizadas por los usuarios finales, y con ello tener la certeza de que la herramienta satisfaga los requerimientos funcionales, por otra parte se recopilará una muestra de 10 proyectos en la ejecución de la herramienta para tener una muestra que compruebe que se disminuyó el esfuerzo asociado a la recolección y análisis de mediciones realizado con anterioridad.

2. Trabajos Relacionados

En el ámbito de desarrollo de software Westfall Linda en [5] propone una serie de pasos para la selección, diseño e implementación de métricas de software posibles de aplicar en un proceso de desarrollo de software de calidad. Estos pasos se centran en la elección de diferentes entidades y atributos que ayuden a medir la calidad de los procesos de desarrollo de software los cuales se dividen en 3 fases, la primera fase indica cómo elegir las diferentes métricas y entidades posibles a tomar en cuenta, la segunda fase describe cómo diseñar y adaptar los indicadores seleccionados a este proceso y por último la tercera fase indica cómo elegir a las personas encargadas de la recolección de datos. En este trabajo relacionado cómo en el que se propone en este artículo son adaptativos al proceso actual de la empresa pero el trabajo de Linda Westfall carece de sustento en algún modelo de procesos y de una herramienta que apoye a su implementación. Por otro lado Pete Rotella y Sunita Chulani en [6], presentan la experiencia adquirida en Cisco System, Inc. en la adopción de un proceso de desarrollo de software de calidad unificado, para controlar sus departamentos y proyectos. Este control se observaba casi imposible ya que cada uno de los departamentos tiene su método de trabajo, para solucionar esto se desarrolló una herramienta llamada SWQD (Software Quality Dashboard, Panel de Calidad de Software), el cual inició con 13 indicadores comunes entre departamentos en el año 2011, y para el año 2012 se integraron entre 16 y 18 indicadores. Esta integración logro aumentar el porcentaje de aceptación entre los clientes de un 30% aumento a un 70%. Si bien este trabajo está enfocado al monitoreo de los diferentes departamentos, no se encuentra basado en modelos de calidad sino en similitudes entre cada uno de los departamentos, es por ello que este control solo es para prácticas específicas de la empresa dejando a un lado el propósito que CMMI-DEV provee para esta área. Otro trabajo interesante es el de Sutherland, J., et al [7], donde se plantea un estudio centrado en empresas de nivel 5 de madurez en CMMI, se propone institucionalizar métodos ágiles en su proceso de desarrollo de software, indicando que el uso selectivo de estos métodos disminuye la indisciplina dentro de las instituciones de desarrollo de software. En este

trabajo se proponen mezclar CMMI-DEV con SCRUM, mediante 12 prácticas genéricas relacionadas con los niveles 2 y 3 de CMMI, esta unión fortalece la adaptabilidad y la previsibilidad de los procesos. El propósito final de este trabajo es detectar errores por medio de una metodología, pero solo se logra adaptando nuevos procesos a lo actual, por lo cual esto genera un trabajo extra además de que quizá sea necesario un departamento especializado en estas prácticas, además de carecer de una herramienta que apoye al funcionamiento esperado, caso contrario al trabajo que se propone en el desarrollo de este artículo el cual es adaptado al proceso actual sin generar trabajo extra. Por otro lado Ferreira, A.I.F., et al en [8], presentan cómo una empresa logra evolucionar a niveles altos de madurez en CMMI, indicando que: La satisfacción del cliente, la mejora de la calidad y la adopción de la reutilización entre otros aspectos, son problemáticas comunes en la búsqueda de esta mejora continua. Se presenta el caso de la empresa BL Informatic's, donde se puso en marcha un plan de mejora que atiende la eliminación de la resistencia al cambio, la reducción de costos en los distintos procesos y áreas de producción, así como la estrategia para aumentar la motivación de sus colaboradores directos e indirectos, todo en base a diferentes normativas como CMMI nivel 3, MPS.BR Nivel F e ISO 9001:2000. Teniendo como resultado al finalizar el proyecto de BL Informatic's, una reducción en sus pérdidas económicas de manera significativa, esto debido a que la identificación de fallas y errores ahora se logra desde la fase de planeación, facilitando así la toma de decisiones con anticipación. Viendo este trabajo se llega a la conclusión que el propósito del proyecto de BL Informatic's está basado en lo esperado con la implementación del área de "Monitorización y Control de Proyectos" de CMMI-DEV pero dejando a un lado el trabajo mediante sus líneas base y límites de confianza. Por otro lado J. Pino, et al [2], muestran que en el mundo existen muchas empresas de desarrollo de software, las cuales son denominadas PyMES (Pequeñas y Medianas Empresas) con no más de 25 empleados, las cuales representan un ingreso económico alto en cualquier país, por ello es necesario que estas empresas tengan una comprensión y amplio conocimiento, en cómo implementar los diferentes estándares propuestos por los Indicadores de Editoriales Académicas (Scholarly Publishers Indicators) correctamente, tanto a nivel internacional como regional, en el desarrollo de proyectos de software. Para esto este trabajo propone una estrategia de selección y priorización de procesos, listando en primera instancia todos los procesos detectados en la empresa, teniendo como objetivo priorizar los procesos mediante las diferentes normativas de CMMI, ISO/IEC 15504, ISO / IEC 12207, entre otras. De igual manera se propone un método que permite la priorización de procesos, este método proporciona diferentes opciones para priorizar, cómo lo es ordenado por: Mejora de grupos de procesos, procesos de gestión, procesos de soporte y por último los procesos de ingeniería. Los autores exponen que la buena práctica de esta metodología, implica una mejora en la calidad de sus procesos de desarrollo de software en las PyMES, y que con el tiempo la productividad en los países que lo implementen logre un aumento significativo. Si bien este trabajo está enfocado en la búsqueda de la mejora continua, no está fielmente basada en una práctica que ayude a este propósito, al contrario del propuesto en este artículo que se basa en prácticas probadas y adaptadas al proceso implementado actualmente,

además de carecer de herramientas que apoyen este funcionamiento de manera automatizada.

Cómo se puede apreciar en los trabajos antes citados se menciona el desarrollo de herramientas de soporte al proceso de desarrollo, sin embargo no se menciona una herramienta específica para la monitorización y control de límites de control de proyectos en una empresa de nivel 5 de madurez del modelo CMMI-DEV. Hasta el momento con la investigación realizada no se conoce alguna herramienta semejante, que sea capaz de llevar este proceso con métricas probadas y establecidas.

3. Descripción de la Herramienta Propuesta

La herramienta propuesta establece los requerimientos mediante los casos de uso numerados a continuación:

1. Consultar Panel de Control
2. Crear Panel de Control
3. Generar Reporte de Gráfico de Control SPI, CPI, Esfuerzo
4. Verificar Productividad de Subprocesos Críticos
5. Generar Reporte de Revisión por Pares
6. Generar Reporte de Productividad y Defectos en Revisiones por Pares
7. Generar Reporte de Variación de Margen Financiero
8. Establecer Límites

En la figura 1 se representa mediante un diagrama de casos de uso, la interacción que tienen los dos actores identificados con cada uno de los casos de uso, estos actores son: “Intellego” y los “Líder de Proyecto”, ambos tienen acceso a los casos de uso de: “Consultar Panel de Control”, el cual proporciona al usuario la posibilidad de verificar o modificar los datos del panel de control que estén asignados a este usuario, su funcionamiento consiste en autocompletar los valores previamente registrados de todos los módulos del sistema. Por otro lado el caso de uso de “Crear Panel de Control” genera espacio de almacenamiento en la base de datos del sistema para un proyecto nuevo, esto para que posterior a ellos se visualicen los indicadores con la información de Jira en cada uno de los módulos para ese panel de control. El caso de uso “Generar Reporte de Gráfico de Control SPI, CPI, Esfuerzo”, da a conocer la tendencia de los indicadores mediante gráficos de control, dependiendo de las revisiones realizadas semanalmente, y en caso de encontrar desviaciones por fuera de los límites de control se lleva a cabo un análisis causal para posterior a ello identificar mejoras que establezcan el comportamiento y el desempeño. Con respecto al caso de uso de “Verificar Productividad de Subprocesos Críticos”, se obtiene la productividad (horas por puntos de función) en operaciones críticas en el desarrollo de software definidas por Intellego y se calcula con base a los datos de esfuerzo en Jira y del modelo de estimación del proyecto para conocer el tamaño del módulo, después se visualizan los datos mediante gráficos de control donde se observa si el comportamiento del proyecto está dentro de los límites de control. Por otro lado en el caso de uso “Generar Re-

porte de Revisión por Pares” se obtienen los datos de los tipos de revisiones realizadas durante el desarrollo de los proyectos, así como defectos, esfuerzo asociado, producto revisado, entre otros y obtener el valor de re trabajo de cada una de ellas. El caso de uso de “Reporte de Productividad y Defectos en Revisiones por Pares”, se da a conocer la productividad en los siguientes tipos de revisiones: 1. Email, 2. Walkthrough, 3. Inspecciones, las cuales se efectúan en los proyectos, así como la densidad de defectos (defectos por cada punto de función), calculado mediante datos de Jira y del modelo de estimación del proyecto. Y por último dentro de los casos de uso compartidos para a los dos tipos de usuarios se encuentra el caso de uso “Generar Reporte de Variación de Margen Financiero” presenta el costos devengados, ingresos a facturar, margen financiero y acumulado financieros generados en periodos semanales. Por último existe un caso de uso que solo el actor "Intellego" tiene acceso, el cual es el de “Establecer Límites” donde se podrán modificar los limites de control y de confianza para cada uno de los proyectos, ya que de inicio se creará el panel de control con valores calculados automáticamente de acuerdo a datos históricos de la empresa. Estos casos de uso en conjunto forman la herramienta propuesta, la cual proporciona indicadores y apoya visualizando tendencias de los proyectos en desarrollo.

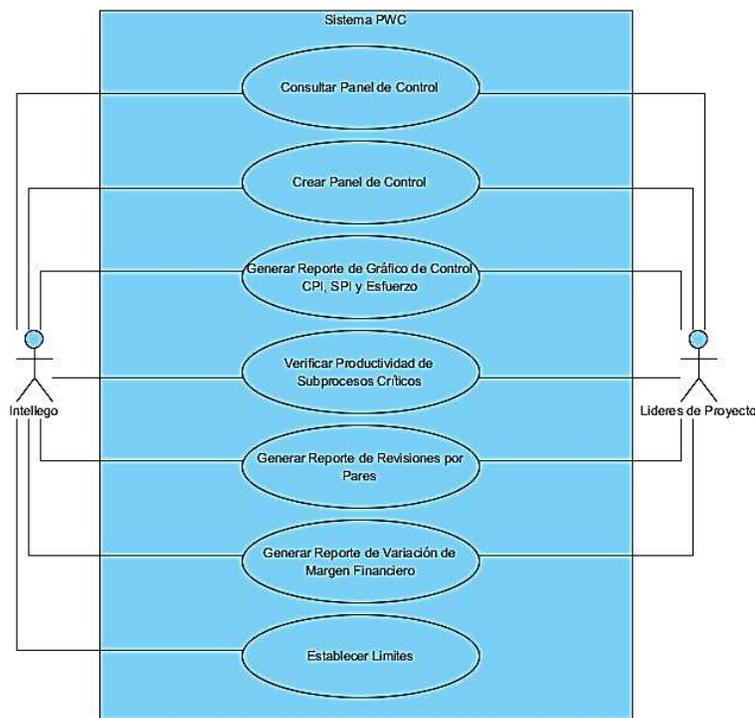


Figura 1. Diagrama de casos de uso – Interacción sobre la herramienta.

3.1 Arquitectura Propuesta

La figura 2 muestra la propuesta de arquitectura describiendo la estructura del software a desarrollar, la cual se basa en el patrón arquitectónico en capas. A continuación se describen cada uno de los elementos que se encuentran en la arquitectura:

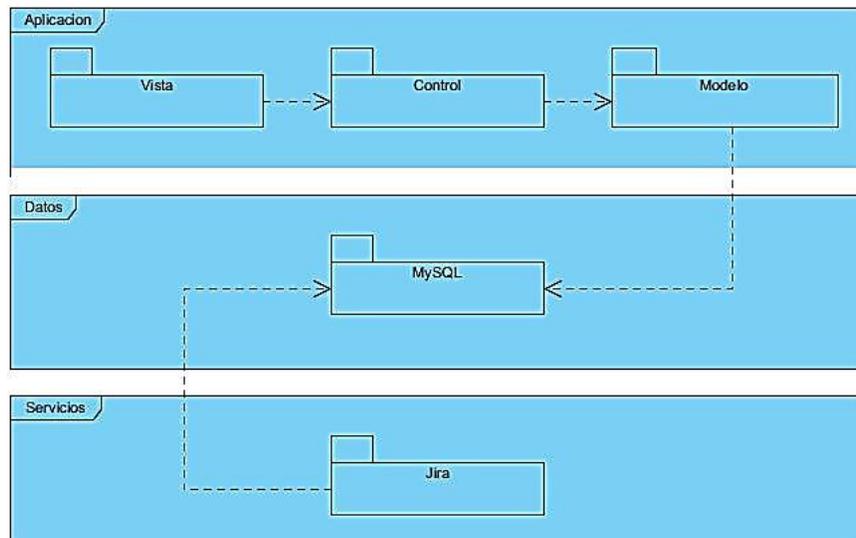


Figura 2. Arquitectura Propuesta.

Capa de Servicios: En esta capa se encuentra instalada la herramienta Jira la cual es un sistema de producción diaria donde se registra el esfuerzo en los procesos y sub procesos de la empresa, esta herramienta se encuentran alineada al nivel 5 del modelo CMMI-DEV, y es gestionada por los ingenieros de Intellego encargados de los proyectos en desarrollo. Además de que esta herramienta se encuentra interconectada a la capa de datos donde almacena la información registrada.

Capa de Datos: Contiene el gestor de base de datos donde se aloja el conjunto de datos generados por el cliente JIRA de cada uno de los proyectos desarrollados y en desarrollo por Intellego, los cuales son gestionados mediante el gestor de base de datos de MySQL. Cabe mencionar que dentro de este gestor existe un componente que contiene los indicadores para satisfacer la herramienta.

Capa de Aplicación: Contiene el código fuente de la herramienta, el cual adopta el patrón arquitectónico MVC (Modelo Vista Controlador). Esta capa depende directamente de la capa de datos, para el consumo de los indicadores necesarios.

En la figura 3 se muestra el diagrama de secuencia UML correspondiente al proceso general de la aplicación propuesta para Intellego, en el cual se identifica el siguiente flujo de trabajo:

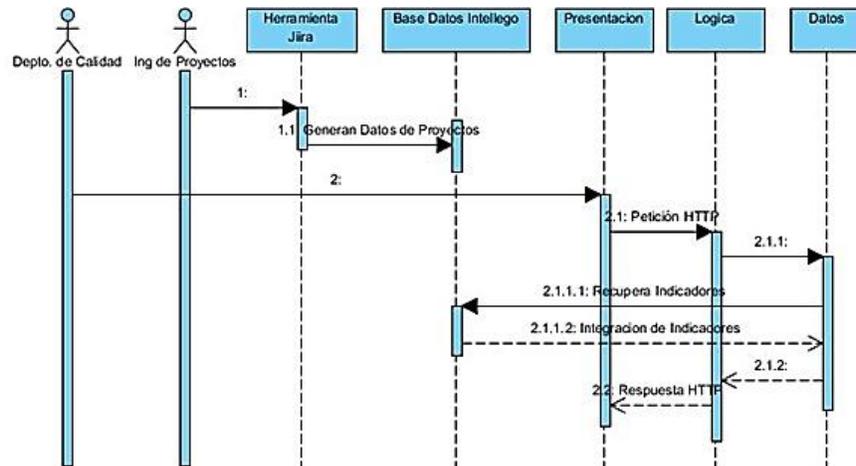


Figura 3. Diagrama de secuencia – Secuencia del proceso general de la aplicación.

1. Los ingenieros de proyectos acceden a la herramienta Jira, para ingresar los datos del proyecto en el que se encuentran laborando en ese momento, esto se lleva a cabo de acuerdo a sus fechas estipuladas.
2. Se genera dentro de la base de datos de Intellego, el registro de los datos del proyecto gestionado.
3. Los encargados de “Intellego” o los “Líderes del proyecto”, podrán ver la información mediante un navegador Web, ingresando a la herramienta en la capa de presentación.
4. El navegador realiza una petición HTTP al controlador de la herramienta, solicitando un proyecto específico, esto dentro de la capa lógica de la herramienta.
5. La capa lógica realiza la petición de los indicadores a la capa de datos, para que esta última retorne a la capa lógica los Indicadores del proyecto seleccionado.
6. Teniendo los indicadores disponibles del proyecto, la capa lógica retorna a la capa de presentación mediante una respuesta.
7. La capa lógica emite una respuesta HTTP con los valores, y con ellos mostrarlos en el navegador Web.

En la figura 4 se muestra un diagrama despliegue representando la integración de la herramienta dentro de la infraestructura actual de Intellego, donde el propósito es observar la distribución de los 3 elementos involucrados que a continuación se describen: Uno de ellos es el “Servidor de datos”, que es donde se encuentra el gestor de base de datos administrando los Indicadores de la empresa generados por JIRA, y

conectado a él mediante un protocolo TCP/IP se encuentra el “Servidor de aplicación”, donde se instalara la herramienta a desarrollar en el lenguaje ASP.Net y por último está el “Cliente”, el cual es considerado cualquier dispositivo que mediante el protocolo TCP/IP tenga acceso a la red de Intellego y que además posea un browser (Navegador Web) instalado, solo con estas características podrá hacer una petición HTTP al servidor de aplicación y así ingresar a la herramienta.

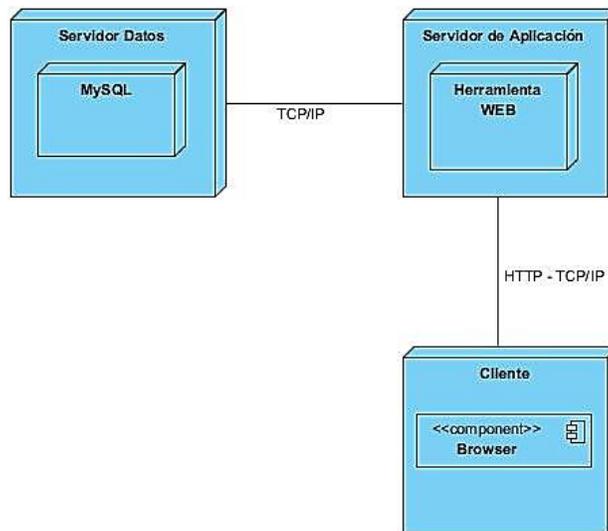


Figura 4. Despliegue de la Herramienta.

3.2 Tecnología a Utilizar

La herramienta se desarrollará en el lenguaje ASP.Net, para integrarla adecuadamente a las tecnologías que maneja Intellego, esta tecnología es la más estable en sus servidores. Además, actualmente ASP.Net soporta 3 modelos de programación: ASP.Net Web Forms, ASP.Net MVC y ASP.Net Web Pages, de los cuales el modelo ASP.Net MVC al implementarse en el desarrollo, separa claramente la lógica de presentación, de negocios y la de acceso a datos.

En el caso del gestor de la base de datos MySQL se establece cómo repositorio propio de la herramienta, además de ser este el consumido por la herramienta JIRA y así facilitar la compatibilidad en la integración de ambas herramientas.

Para el desarrollo de la herramienta se implementa el ciclo de vida iterativo incremental, debido a que permite realizar el ciclo completo del proyecto, y en cada iteración se permite añadir funcionalidad o refinar los requerimientos, además se implementara la metodología Scrum que permite priorizar funcionalidades y resolver dudas del alcance con colaboración del "Product Owner", permitiendo así un desarrollo ágil.

Por otra parte en cada iteración se hará un caso de uso de la herramienta, y al finalizar cada uno de ellos se realizará las pruebas, además en el proyecto se elaborarán todos los entregables estipulados por la empresa Intellego.

4. Trabajo a Futuro

Para la continuación del desarrollo de la herramienta, es necesario analizar la estructura de la base de datos de Intellego para recuperar la información requerida para visualizar los indicadores que provee el sistema. Teniendo este análisis inicia la construcción de los módulos que sustentara las especificaciones del área, mediante los requerimientos funcionales: 1. Consultar Panel de Control, 2. Crear Panel de Control, 3. Generar Reporte de Gráfico de Control SPI, CPI, Esfuerzo, 4. Verificar Productividad de Subprocesos Críticos, 5. Generar Reporte de Revisión por Pares, 6. Generar Reporte de Productividad y Defectos en Revisiones por Pares, 7. Generar Reporte de Variación de Margen Financiero y 8. Establecer Límites, además conforme se reporten los avances de la herramienta generar los entregables enfocados al nivel de madurez de la empresa Intellego.

Una vez construida la herramienta se procederá al pilotaje en su totalidad de la herramienta a fin de tener una muestra significativa de proyectos que permitan analizar el ahorro de esfuerzo obtenido y comparar respecto a la meta planteada.

5. Conclusiones

Finalizando esta etapa de desarrollo de la herramienta, se observa que el monitoreo de los proyectos que se tiene actualmente mediante plantillas en Excel, si bien abarca el funcionamiento requerido para llevar a cabo esta actividad, no disminuye el trabajo realizado, ya que la recolección de los diferentes indicadores de la interfaz de JIRA genera un trabajo extra en sus actividades normales, por ello la herramienta se considera viable para apoyar el funcionamiento del área de CMMI-DEV llamada "Monitorización y Control de Proyectos" y con ello reducir el esfuerzo en un 50% de actual.

Agradecimientos

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca otorgada para la realización de los estudios de Maestría, y a la empresa Intellego por su disponibilidad en información y por el apoyo brindado para la publicación de este artículo.

Referencias

1. Intellego. <http://www.intellego.com.mx/country-selector>. 09-Abril-2013.
2. Francisco J. Pino, Félix García, Mario Piattin. Key processes to start software process improvement in small companies. Proceeding SAC '09 Proceedings of the 2009 ACM symposium on Applied Computing, 2009, PP: 509-516.
3. Olga Liliana Castrillón Giraldo, Gustavo Adolfo Mina Valencia. Framework para gestión de conocimiento de equipos de desarrollo de software basado el modelo CMMI DEV 1.3. Colombia, 2011.
4. Mike Konrad, Sandy Shrum. CMMI (2a. Ed.): Guía para la integración de procesos y la mejora de productos, Addison-Wesley, 2009.
5. Linda Westfall. 12 Steps to Useful Software Metrics. The Westfall Team, 2005.
6. Pete Rotella, Sunita Chulani. Implementing quality metrics and goals at the corporate level. Proceeding MSR '11 Proceedings of the 8th Working Conference on Mining Software Repositories, 2011, PP: 113 - 122.
7. Jeff Sutherland, Carsten Ruseng Jakobsen, Kent Johnson. Scrum and CMMI Level 5: The Magic Potion for Code Warriors. Proceeding HICSS '08 Proceedings of the Proceedings of the 41st Annual Hawaii International Conference on System Sciences, 2008, PP: 466.
8. Analia Irigoyen Ferreiro Ferreira, Gleison Santos, Roberta Cerqueira, Mariano Montoni, Ahilton Barreto, Andrea O. Soares Barreto, Ana Regina Rocha. Applying ISO 9001: 2000, MPS.BR and CMMI to Achieve Software Process Maturity: BL Informatica's Pathway. Proceeding ICSE '07 Proceedings of the 29th international conference on Software Engineering, 2007, PP: 642 - 651.

Security in Web applications

A. Santoyo-Sanchez¹, C. De Jesús-Velásquez²,
L. I. Aguirre-Salas³

¹Department of Computing, Universidad de Guadalajara –CUCEI, Guadalajara, Jalisco, México

²Compatibility Validation, Intel Tecnología de México S.A., Tlaquepaque, Jalisco, México

³Department of Engineering, Universidad de Guadalajara – CUCSUR, Autlán de Navarro,
Jalisco, México

alejandra.santoyo@cucei.udg.mx

Abstract. The scientific and business communities today are particularly concerned to ensure that web applications containing less vulnerability on the issues of security and reliability of the information. This article proposes a guide with the aim of guiding the engineers Web application testing to find vulnerabilities within an application. This guide is based on standards ISO 9126:1991, IEEE 1233, IEEE 6190, IEEE 830, ISO / IEC 9646-1. The guide is currently incorporated in a wizard; some of the practical results obtained from using the same are presented in the case study.

Keywords:WEB application, quality, reliability, security, vulnerability.

1. Introduction

The use of Web applications to conduct business, banking, educational, information, e-business, issuance of certificates, among others, has prompted interest in ensuring the safety and reliability of the information displayed, processed, stored and/or transferred via the web.

The safety and reliability of Web applications has been approached from different perspectives and with different methods. Among the solutions are formal methods [1] and [2] which focus on specify, develop and test the quality of computer systems by implementing a rigorous mathematical notation. Unfortunately, formal methods are not generally used by the difficulty in practice in the use of such models.

In industry, semi-formal methods [3] [4] and informal [5], [6] are used for modeling and verifying aspects of security and reliability of Web applications due to its easy representation, however models obtained are difficult to analyze what aspects ensure security and reliability is a complex and imprecise. As different research groups have been building bridges between formal methods, semi-formal and informal as in [7].

Moreover according to [8] software vulnerabilities can be grouped into three categories: the design, development and implementation, and operation. Unfortunately, in practice, analysts and designers of Web applications do not specify security requirements, and often do not provide for vulnerabilities in their design.

This represents a challenge in the field on Information Technology Audit. So, for that reason in order to avoid these problematic this paper proposes a set of guidelines based on standards: ISO 9126:1991, IEEE 1233, IEEE 6190, IEEE 830, ISO / IEC 9646-1 which will be used in the testing stage of a development process of WEB software systems to help engineers in reducing the vulnerabilities of Web applications during the testing phase.

This paper is organized as follows. Section 2 describes how the security in Web application has addressed. Section 3 presents the contribution of this paper, i.e. the proposal guidance to achieve security and reliability within a Web application; whereas the section 4 presents a case study to illustrate the use of the guide. Finally, section 5 provides conclusions and future work.

2. Background

A Web application consists of a set of Web pages and components that interact to form a system running using server (s) Web, network (s), HTTP, and browser, in which its state changes to the information that users enter or request [9].

For quality Web applications from development methodologies, there are two main approaches 1) hypermedia community, and 2) the software engineering community [3]. The first takes care of all the methods and guidelines for writing, designing or writing content for Web publishing [10].

While the software engineering community in Web Engineering addresses the application of systematic methodologies, structured and measurable to develop, test and maintain Web applications [11] [12].

Unfortunately Web engineering has focused on the phases of life cycle analysis and design [13], while Web applications in addition require experts in the field of security to ensure data integrity, reliability, etc. [14].

According to ISO/IEC17799 [15] indicates that the security means a set of methodologies, practices and procedures that seek to protect information as a valuable asset and thereby reduce threats and risks (called information security) to ensure that resources system to be used the way it was decided (called security). In IEEE/std 610.12 [16] reliability is the ability of a system to perform the functions for which it was designed without failure, in other words intended to ensure that the information is not modified by anyone not authorized.

But despite numerous efforts to achieve security through code review and practice of software engineering, many professionally designed Web sites still suffer from security holes and make serious applications. This evidence suggests a need for tools and techniques to address the problem [17] [18].

Methods of developing security for the Web is a daunting task, partly because of security concerns arose after the development of the web application or after a security problem occurred. In [19] describes various tools that can be used for security within a Web application using a test, some of which are WEBGOAT, NETCAT, CURL and PROXIES. WEBGOAT created by OWASP (Open Web Application Security Project) is a Web application developed in J2EE designed with the objective of serving as a test bed and find vulnerabilities once Web application is deployed, it is supported by other tools like NETCAT which used to interact at http or any other protocol with the Web server, while CURL interacts at http command. Local PROXIES are used to intercept, modify and send requests to the server. Also looking for vulnerabilities or risk factors through the code, we can use a local copy of the web site. Additionally, in [19] is mentioned for these tools (NETCAT, CURL and proxies) to in order to work properly Firefox browser is required, these tools look for vulnerabilities simple as if you use the right browser, the right platform for the type of Web application, good 80/443 port operation (allowing http from any source, etc.). however these are used when the Web application is already on the server, which is why further investigations are focusing on addressing vulnerabilities in the design stage, although there are some mechanisms to verify single types of vulnerabilities, is not usually implemented if you try to avoid all the vulnerabilities that may appear in your web application, that is the reason we use [1] that describes a methodology for finding vulnerabilities in the design stage.

According to [1] AppScan DE is a method that looks for common vulnerabilities in Web applications and makes a list of them. Subsequently makes recommendations for solutions to ensure safety guidelines for reliability and quality in the application which focuses on UNIX platforms. Using ISO 9126:1991, IEEE 1233, IEEE 6190, IEEE 830, ISO / IEC 9646-1 and information on AppScan DE in the next section introduces the guide.

3. Guide on Finding Vulnerabilities

Vulnerabilities in Web applications in accordance with ISO / IEC 13335-1:2004 in [20] are a group of weaknesses that can be attacked by one or more threats, the most common according to the ISO 9126:1991 in [21], [22] are:

- *Usability*. It focuses on testing the effort needed for use, and individual assessment of such use by a system of users stated or implied.
- *Functionality*. Try the product's ability to provide features that meet specific needs.

- *Reliability*. Refers to test the ability of a product to maintain the level of performance under stated conditions for a period of time.
- *Efficiency*. Try the qualities that have to do with the behavior in time, resource utilization efficiency and compliance.
- *Portability*. Test focuses on the characteristics of adaptability of various settings, and instability co-existence with other platforms.
- *Maintainability*. Refers to prove the qualities of capacity for change, testing facility, stability and ease of maintenance.

On the other hand in [1] found a "perversion" of Web applications illustrated in Table 1, which shows which covers the tool AppScan DE yet within Windows platforms are others like that here are in Table 2.

The guide is based on questionnaires which indicate at each stage of development of a Web application security issues that must ensure the application in each stage. In this way the guide is intended to indicate that it has been validated.

Specifically in the stage of the questionnaires requirements help ensure the detailed definition in the life cycle of web application, where security is inherent.

Table 1. WEB "Perversion" that ensuring AppScan DE.

VULNERABILITY	HARDENED SERVERS	NETWORK SCANNERS	ACCESS CONTROL	SECURITY OF WEB APPLICATION WITH APPSCAN DE
HIDDEN HANDLING				✓
COOKIE POISONING				✓
BACKDOORS				✓
BUFFER OVERFLOW	✓			✓
STEALTH COMMANDS				✓
KNOWN VULNERABILITIES	✓	✓		✓
MY SETTINGS THIRD	✓	✓		✓
MANIPULATING PARAMETERS				✓
CROSS SITE SCRIPTING				✓
FORCED BROWSING			✓	✓

Table 2. Main issues of vulnerability found in Windows platforms.

WINDOWS SECURITY PROBLEMS	RESULT
PROTECTION OF POTENTIAL CUSTOMERS AND USERS OF VIRUSES THAT ENTER THROUGH THE VULNERABILITY IN THE WINDOWS META FILE CODE (WMF).	THE SECURITY FLAW BY DOWNLOADING AN INFECTED FILE ALLOWS A HACKER TO TAKE CONTROL OF YOUR PC, STEAL INFORMATION, PERSONAL PASSWORDS, ETC.
INTERNET EXPLORER ALLOWS REMOTE CODE EXECUTION.	ALLOWS THE VULNERABILITY IN THE BROWSER FOR REMOTE CODE EXECUTION.
UPDATES TO SNMP.	VULNERABILITY ALLOWS SIMPLE NETWORK MANAGEMENT PROTOCOL THAT REMOTE CODE EXECUTION.
INCREASED PRIVILEGES.	ALLOWS VULNERABILITY THAT ALLOWS PRIVILEGE ESCALATION.
INTERNET EXPLORER, ACCESS TO DOCUMENTS IN ANOTHER PAGE.	ERROR ON HOW TO HANDLE REDIRECTS THE URLS IN HTML MANAGER, WHICH ALLOWS ACCESSING DOCUMENTS SERVED FROM ANOTHER WEB PAGE.

In addition to ensuring they are implemented in the security policy and standards for the development team.

Some questions contained in the questionnaire are:

1. Is there a security protocol to avoid outside attacks and intrusions from hackers?
2. Do you avoid being seen, the program names and directories?
3. Does it protect the integrity of their programs and data?
4. Are the services offered are made via secure transaction channels?

Just to mention some of the questions you can ask in order to have a general idea of the aspects to be taken into account for accomplishing security designing a Web application.

In *analysis* and *design* phases looking for inconsistencies in design requirements, we recommend taking into account the following security mechanisms: user management, authentication, authorization, data confidentiality, integrity, reliability, session management, transport security, segregation system in levels, and privacy.

During *implementation* phase coding is completed in the design of a Web application. The goal is to generate a list of recommendations for developing and testing the sequence relationships between webpages in order to get a good robust application and ensure optimum performance. Issues that must meet a Web application according to ISO 9126:1991 with IEC (International Electrotechnical Commission) [21] and IEEE / Std 610.12, 1990 [16] are functionality, reliability and usability.

In *test* phase should check the requirements, analyze the design and code review. In case of errors or new requirements in Web development requires a budget to determine the cost, time and impact of a change in the existing product, document the change and verify the consistency changes.

As the test engineer must first identify all possible security holes site considering the characteristics tested. The issues proposed in this article for safety testing are as follows.

- 1) *DNS management*: Its main purpose is to detect whether the domain name is appropriate. More than one domain name can be assigned to a Web site. It is possible to add other names under the **.MX** or another, which will generate additional aliases for the site. It is recommended that all domains are redirected to the first screen corresponds to the official homepage of the Website.
- 2) *Protection of internal structure of the Website*: In this case you should check the site's internal structure, i.e. reducing the amount of information contained in the URL shown in the program as user data display, and directory names program, etc. It is recommended that mechanisms for transferring information between webpages is at the level of server objects, thereby preventing the customer to be responsible for the transfer of data between the server running sessions. Another way is to prevent access to elements of Web server addresses is associated with a session on or associated with *SessionId* or *UserId*, this because with simple steps you can know the session *token* and pretend that this is the same user returning to the site. Protections should be incorporated on the address source IP address.
- 3) *Protection against robots*: Determine if the search robots or spiders known as spiders from entering them. Not all site directories should be available for search bots to enter them. You must use the *robots.txt* file, which is a plain text file containing instructions in the meta-tags of the homepage to prevent their access.
- 4) *Managing privacy*: verify the privacy of users of the site permanently, according to site policy. There should be physical and logical protections on information. For example customers protect physical servers in different data storage, ideally separate interfaces including data query. Also incorporate mechanisms for data encryption to sensitive information.
- 5) *Secure channel*: Verify the encryption of the communication channel for transferring private information between users and the Web site, via the Internet. It recommends the establishment of the channel SSL (Secure Socket

Layer), although the mean delay in starting the initial connection, not subsequently results in increased bandwidth and server resources increases.

- 6) *Access control mechanisms*: Check the protection of user privacy on the web site content, in the context of keys and authentication as simple and advanced electronic signature which is a system that identifies the user when performing transactions through Internet or closed networks, another way is through authentication with username and password pair that should provide key feedback mechanisms, by providing answers to predefined users, and do so by email. You can also use hardware systems for authentication, in this case should incorporate mechanisms such as token cards, and security.
- 7) *Protection program*: Check the protection code and the server's internal programs. How to avoid the transfer of parameters across the direction of access to the pages, to avoid reading the executable from the directory server, in the case of scripts using code compaction.
- 8) *External versus self-Site Hosting*: an objective evaluation based on capabilities, support requirements and effective.
- 9) *To ensure minimal roles*: defining the various roles within the definition and design of the site. That is, choose the most competent staff that can meet them, for example, the architect who is responsible for making working configurations of servers and applications, the application manager, the quality control manager, security manager, among others.

Table 3 is a brief questionnaire that helps the verification of vulnerabilities and compliance.

To carry out a general and adequate guidance is recommended the following practices.

- *Test early and test often*: This practice focuses on testing before, during and after the Web application in which these tests should be performed frequently.
- *Understand the scope of security*: Classify information according to the degree of protection considering legal issues.
- *Understand the scope of study*: Having accurate documentation as architecture, data flow diagrams, use cases, etc., minimum infrastructure for monitoring and analysis of attacks and network applications.
- *Use the tools*: there are open source and commercial, they simplify and speed the security process, helping the security personnel in their tasks.
- *What is important is in the details*: Review results and eliminate errors.
- *Use the source code when available*: check the code to assist in this review and discover the vulnerabilities that are not detected in the inspection of code phases.

4. Case of Study

The guide was used and tested during the development of a Web application for a company, but for reasons of confidentiality name is omitted.

The aim of the guide is to make relevant suggestions about safety and reliability in the testing process of Web applications on Windows platforms in a way that ensures the integrity and proper functioning of them. This guide is done in phases and each phase of life cycle contains the criteria that indicate when you can go from one phase to another.

Table 3. Format verification of security testing

SECURITY CONCEPTS	MEETS	
	YES	NO
DOES THE SITE WORK PROPERLY AND NO FAILURES TO BROWSE THEIR PAGES OR USE YOUR SERVICES? (ESPECIALLY IN THE CASE OF ONLINE TRANSACTIONS).		
THE DATA ENTERED BY A USER THROUGH FORMS; ARE VALIDATED BEFORE BEING SENT AND PROCESSED BY THE SITE SERVER?		
IS THERE A SECURITY PROTOCOL TO AVOID OUTSIDE ATTACKS AND INTRUSIONS FROM HACKERS?		

A. Inception.

In order to use the guide in the test site must have certain characteristics to determine whether or not, using the following evaluation criteria:

- Attendance of all involved to define the cost / time and scope.
- Agreement that the requirements have been established and captured, and there is a shared understanding of these.
- Agreement that the estimate of the cost / time priority, the risks and the development process are appropriate.
- All risks are identified and a mitigation strategy is each.

B. Elaboration.

At the end of the design phase objectives were examined and detailed scope of the system, the choice of architecture, and resolution of major risks.

C. Construction.

Once the product was ready for delivery it was necessary to do the following: ensure that all functionality has been developed and all tests are completed.

D. Safety Test.

The activities you can do to make safety tests are diverse and are directed to several areas in the tables 4 to 12.

The guide will enable and disable points to prove. After you define the effort devoted to each test, i.e. testing effort goals are conducted to verify the safety and reliability of the Web application.

Table 4. DNS test Management

DNS MANAGEMENT	MEETS	
	YES	NO
DO ALL SITE LINKS ARE ASSOCIATED PAGE AND APPROPRIATE CONTENT TO LINK SAID?		
IF A SEARCH WITHIN THE SITE OR ANY OPERATION THEREIN, ARE THE RESULTS DISPLAYED PROPERLY?		

Access requirements. Site users must have access to the intranet and database.

Table 5. Protection of internal structure of the Web site

PROTECTION OF INTERNAL STRUCTURE OF THE WEB SITE	MEETS	
	YES	NO
DO YOU AVOID BEING SEEN THE NAMES OF THE PROGRAMS AND DIRECTORIES?		
DO THEY TAKE STEPS TO CONFIRM THAT THE SAME USER RETURNS TO THE SITE?		

Table 6. Protection against robots

PROTECTION AGAINST ROBOTS	MEETS	
	YES	NO
IS IT PREVENTS ACCESS TO PROTECTED DIRECTORIES?		
DO META-TAGS USED ARE APPROPRIATE TO PROTECT THE SITE'S DIRECTORY?		

Table 7. Privacy Management Test

PRIVACY MANAGEMENT	MEETS	
	Yes	NO
ARE PRIVATE DATA, PROVIDED VOLUNTARILY BY USERS ARE STORED IN A RESERVED MANNER?		
DO YOU OFFER A PRIVACY POLICY OF PERSONAL DATA AND REPORTED ITS EXISTENCE IN THE RELEVANT PAGES?		
DOES IT PROTECT THE INTEGRITY OF THEIR PROGRAMS AND DATA?		

Table 8. Secure Channels Test

SECURE CHANNEL	MEETS	
	Yes	NO
ARE THE SERVICES OFFERED ARE MADE VIA SECURE TRANSACTION CHANNELS?		
IS INCORPORATED ENCRYPTION MECHANISMS OF THE COMMUNICATION CHANNEL FOR TRANSFERRING PRIVATE INFORMATION BETWEEN USERS AND THE WEBSITE? (SSL, PGP, ETC.).		

Table 9. Access Control Mechanisms Test

ACCESS CONTROL MECHANISMS	MEETS	
	Yes	NO
DOES THE SITE WORK PROPERLY AND NO FAILURES TO BROWSE THEIR PAGES OR USE YOUR SERVICES? (ESPECIALLY IN THE CASE OF ONLINE TRANSACTIONS)		
ARE THE DATA ENTERED BY A USER THROUGH FORMS VALIDATED BEFORE BEING SENT AND PROCESSED BY THE SITE SERVER?		
HOW THE ISSUES THAT REQUIRE RESTRICTED ACCESS, THE SITE PROVIDES A MEANS TO VALIDATE ACCESS, I.E. THROUGH A BOX WITH USERNAME AND PASSWORD?		

Table 10. Protection programs Test

PROTECTION PROGRAMS	MEETS	
	YES	NO
IS THERE A SECURITY PROTOCOL TO AVOID OUTSIDE ATTACKS AND INTRUSIONS FROM HACKERS?		
DO YOU HAVE A DATA BACKUP POLICY IN ORDER TO OVERCOME EFFECTS OF FAILURES DERIVED FROM THE PREVIOUS POINT?		

Table 11. External versus self-Site Hosting Test

EXTERNAL VERSUS SELF-SITE HOSTING	MEETS	
	YES	NO
IS ALLOWS COMBINATION OF SERVICES AND INFRASTRUCTURE TO PROVIDE AN OPTIMAL SOLUTION SET IN TERMS OF COST-BENEFT?		
IS AN ASSESSMENT OF THE OVERALL IMPACT OF ALL ELEMENTS INVOLVED IN THE SITE?		

Table 12. Ensure minimal roles Test

ENSURE MINIMAL ROLES	MEETS	
	YES	NO
DOES THE SECURITY POLICY IMPLEMENTED TO VALIDATE THE RESTRICTED ACCESS IS ADEQUATE TO THE PURPOSES OF SERVICE OR THE INSTITUTION?		
IN THE CONTEXT OF WEB SITE OPERATION: DO YOU HAVE STAFF ABLE TO MEET THE DIVERSE FUNCTIONAL ROLES?		

Table 13. Type of test recommended

TYPE OF TEST	APPLIES	
	YES	NO
DNS MANAGEMENT		
PROTECTING YOUR WEB SITE'S INTERNAL STRUCTURE		
PROTECTION AGAINST ROBOTS		
PRIVACY MANAGEMENT		
SECURE CHANNELS		
ACCESS CONTROL MECHANISMS		
PROTECTION PROGRAM		
EXTERNAL HOSTING VS. OWN SITE		
ENSURE MINIMUM ROLES		

Resources for the effort test. For example: *HARDWARE*. Any computer with Internet access, network infrastructure at least 100 Mbps. *SOFTWARE*: Platform Windows installed with any browser. Or of *PERSONAL*: test engineer, application support (on request).

Table 13 shows the results after a series of tests with which ensure quality and application security.

5. Conclusions

This paper has presented a guide for test engineers in order to identify vulnerabilities in Web applications, i.e. reliability and safety issues to consider and we'll fix it before delivering the final product to the user, this is achieved by implementing a wizard in which based on responses from online questionnaires related to aspects of safety and reliability of Web applications being tested, the questionnaires are based on ISO 9126:1991 standards, IEEE 1233, IEEE 6190, IEEE 830, ISO / IEC 9646-1.

The usefulness of this method has been demonstrated in a case study, favorable results were obtained in the testing stage and help test engineers to see more clearly where there is more uncertainty in Web applications, also reduces the costs involved fees and better yet, it is easy to use, thus reducing training time cost.

It is important to mention that it is based on results that although sounds outdated, the test guide can be applied to current web application development. Since it is based on common practices, which every developer should know and apply.

As future work, the idea is to implement more projects in a wizard application, so you can find best practices, and according to their results being used as a base for projects more complex.

Acknowledgment

This work was partially supported by SEP in the scope of the project COECYT - Michoacán and ITSCH DELFIN 2008 and 2009.

References

1. C. A. Jerez Lugo. Seguridad para lograr Confiabilidad y Calidad de los Servicios Digitales en Internet. Bachelor dissertation, Ed. Universidad de las Américas: Departamento de Ingeniería en Sistemas Computacionales, Mayo 2004.
2. S. Debnath, P. Mitra, N. Pal, C. Lee Giles. Automatic Identification of Informative Sections of Web Pages. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 9, pp. 1233-1246, 2005.
3. S.M. Abrahão, O.Pastor, L. Olsina, J.J. Fons. Un método para medir el tamaño funcional y evaluar la calidad de sitios Web. IV Jornadas de Ingeniería del Software y Bases de Datos (JISBD), 2001, Almagro (Ciudad Real), Spain, pp. 477 – 490.
4. C.H. Liu Kung, P. Hsia, C.T. Hsu. Structural testing of WEB applications. Proc. 11th IEEE International Symposium on Software Reliability Engineering ISSRE, 2000, San Jose, CA, USA, pp. 84 – 96.
5. S. Zarei. Electronic Service Quality Evaluation Methods for Online-Banking System. International Journal of Computer Science and Technology (IJCST), Vol. 1, Issue 2, pp. 6 – 13, Ed. C/O Ayushman Technologies, 2010.
6. Y. Deshpande, S. Hansen. Web Engineering: Creating a Discipline among Disciplines. Vol.8, No. 2, pp.82 – 87, Ed. IEEE Multimedia, Abril – Junio 2001.
7. D. Andrés Silva, B. Mercerat. Construyendo aplicaciones web con una metodología de diseño orientada a objetos. Megazine: Colombian Journal of Computation, Vol. 2, pp. 79 – 95, Dic. 2001.
8. Desarrollo Seguro de Aplicaciones. <http://es.scribd.com/doc/54453491/6/Aplicaciones-inseguras>, May 2011.
9. S. Sampath, V. Mihaylov, A. Souter, L. Pollock. Composing a framework to automate testing of operational Web-based software. Proc. 20th IEEE International Conference on Software Maintenance (ICSM), 2004, Chicago, IL, USA, pp. 104 – 113.
10. L. Rosefeldt, M. Peter. Information architecture for the world wide Web. Cambridge, Mass, Ed. O'Reilly & Associates, 2002.
11. D. Lowe, B. Henderson-Sellers. Impacts on the development process of differences between web systems and conventional software systems. International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet (SSGRR), 2001, L'Aquila, Italy, pp. 1 – 12.

12. Y. Deshpande, S. Marugesan, A. Ginige, S. Hanse, D. Schawabe, M. Gaedke, and B. White, Web Engineering. *J. Web Eng.*, Vol. 1, No. 1, pp. 3 – 17, 2002.
13. M. Jose Escalona, G. Aragón. NDT. A Model-Driven Approach for Web Requirements. *IEEE Transactions on Software Engineering*, Vol. 34, No. 3, pp. 377 – 390, May-June 2008.
14. Y. Deshpande, S. Hansen. Web Engineering: Creating a Discipline among Disciplines. *IEEE Multimedia*, Vol.8, pp. 82 – 87, April-June 2001.
15. ISO/IEC 17799 (International Standar ISO/IEC 17799 Second edition 2005-06-15) <http://www.iso17799software.com/>.
16. IEEE/std610.12 (Standard Glossary of Software Engineering Terminology) 1990, IEEE <http://www.swen.uwaterloo.ca/~bpekilis/public/softwareEnGlossary.pdf>.
17. A.D. Rubin, D.E. Geer Jr., A survey of WEB security. *IEEE Computer Society Press*, Vol. 31, pp. 34 – 41, Sep 1998.
18. D. Scott, R. Sharp. Developing secure WEB applications. *IEEE Educational Activities Department*, Vol. 6, pp.38 – 45, Nov. 2002.
19. H. M. Racciatti. Seguridad en aplicaciones Web. *Revista @rroba*, Vol. 96, Suplemento Hack paso a paso No.27, Noviembre 2007.
20. <http://www.iso.org>.
21. ISO 9126 (Software Engineering Product Quality part 1, 2, 3), <http://www.iso.org>.
22. J. L. García Cerpas. Sistema asistencial en el proceso de pruebas para microprocesadores y tarjetas madres. Master disertation, Ed. Universidad de Guadalajara Maestría en Sistemas de Información, Julio 2007.