

## Modelo para el desarrollo integrado de aplicaciones compuestas

Orestes Febles Díaz<sup>1</sup>

<sup>1</sup>Universidad de las Ciencias de la Información, La Habana, Cuba. Ponencia enviada al 1er Congreso Científico Internacional Tecnología, Universidad y Sociedad, organizado por la Universidad Central del Este (7-9 de noviembre de 2012).

[ofebles@uci.cu](mailto:ofebles@uci.cu)

Recibido: 10 dic. 2012

Aceptado: 18 feb. 2013

---

### RESUMEN

Se presentó un modelo integrado que representa diferentes componentes de carácter tecnológico, de conocimiento y sus interrelaciones con un componente de selección inteligente de información que apoya la toma de decisiones en el proceso de composición de aplicaciones en entornos orientados a servicios. Este modelo aporta interoperabilidad promoviendo el uso de estándares en el desarrollo integral de aplicaciones compuestas en iniciativas orientadas a servicios.

**PALABRAS CLAVE:** Arquitectura orientada a servicios; MIDAC; Aplicaciones compuestas

### ABSTRACT

**Integrated Development of Composite Applications Model.** It presented an integrated model that represents different components of a technological, knowledge and their interrelationships with a smart selection component of information that supports decision-making in the process of writing applications in service-oriented environments. This model provides interoperability by promoting the use of standards in the development of composite applications in service-oriented initiatives.

**KEYWORDS:** Service-oriented architecture; IDCAM; Composite applications

---

### INTRODUCCIÓN

La ciencia y la tecnología son factores fundamentales en el desarrollo de las sociedades actuales. La aparición de Internet y las comunicaciones de alta velocidad son ejemplos palpables que han transformado de manera significativa la forma de manejar la información. Paralelo a su evolución, han surgido nuevos negocios y con ellos organizaciones que buscan responder de una manera ágil a las crecientes necesidades del mercado. Las organizaciones se enfrentan al reto de aumentar su agilidad de innovación y respuesta a los constantes cambios en los negocios y a los ambientes tecnológicos heterogéneos con la misión de reducir los gastos en sus Tecnologías de la Información (TI).

Una de las dificultades que existen en el desarrollo de aplicaciones informáticas es la falta de interoperabilidad entre sistemas ante las crecientes necesidades de integración. La alineación entre el negocio y las TI constituye uno de los beneficios más importantes que se le atribuyen a la Arquitectura Orientada a Servicios (SOA por sus siglas en inglés). SOA ha revolucionado principalmente la manera de adoptar la tecnología; ha surgido como un paradigma capaz de soportar la agilidad de los procesos de negocio, aumentando la eficacia y eficiencia de las operaciones de las empresas en el mundo de hoy (Matsubara y otros, 2009).

Al adoptar SOA se agregan beneficios referentes a la creación de servicios y aplicaciones que coexisten independientemente de la variedad tecnológica brindándole mayor reusabilidad al conjunto de funcionalidades existentes en la organización. El poder de combinación, composición y reutilización de los servicios estimula la construcción de aplicaciones compuestas que ofrecen soluciones flexibles a las necesidades de la empresa.

Si bien la construcción de una aplicación compuesta integral es una necesidad para las organizaciones en la actualidad, su desarrollo, despliegue y mantenimiento son tareas mucho más complejas (Duggan, 2012). La

variedad de plataformas, herramientas y modelos arquitectónicos necesarios para la construcción de aplicaciones introducen una complejidad técnica que puede convertirse en algo abrumador, pues es necesario tener amplios conocimientos y habilidades para enfrentar este reto. De no tomarse las decisiones correctas pudiera conducir a problemas en el rendimiento, el funcionamiento, la calidad de los servicios y la elevación de los costos de desarrollo y mantenimiento.

Basado en el estudio realizado sobre las aplicaciones compuestas en iniciativas orientadas a servicios es posible destacar las siguientes dificultades y/o limitaciones:

- Las metodologías de desarrollo y marcos de trabajo de las empresas líderes vienen atadas a sus propias herramientas y tecnologías las cuales son vendidas a elevados precios.
- En las fuentes consultadas generalmente se trata la implantación de SOA sobre la base de componer aplicaciones entre servicios y por lo general no se refiere a cómo brindar los beneficios de la composición a desarrolladores y al usuario final, ni cómo apoyar la toma de decisiones (Juric y Krizevnik, 2010).
- La existencia de un número elevado de tecnologías y marcos de trabajo para el desarrollo de aplicaciones compuestas e interfaces enriquecidas dificulta la toma de decisiones en su proceso de selección.
- Existen repositorios para almacenar los recursos (servicios) que pueden ser reutilizados y que contienen información útil sobre los mismos, pero el proceso de selección muchas veces se hace complejo y ambiguo (la información no es clara y/o precisa) (Mohebi, 2012).
- Las organizaciones por lo general no registran las mejores experiencias en el desarrollo de aplicaciones principalmente por la ausencia de herramientas que faciliten la gestión de conocimiento
- Gran parte de estos problemas están asociados a que las organizaciones generalmente no poseen una infraestructura tecnológica adecuada para soportar la interoperabilidad entre aplicaciones, así como el intercambio de información de una manera estandarizada a través de servicios lo que dificulta el desarrollo de aplicaciones compuestas.

Esta problemática influye en que, con frecuencia, se hace complejo el desarrollo de aplicaciones compuestas, por lo que la misma se ha tomado en cuenta para la elaboración del modelo que se propone como parte de la investigación desarrollada.

## **METODOLOGÍA**

Se compararon los modelos actuales más utilizados en la composición de aplicaciones y se realizó un diagnóstico en instituciones productoras de *software* con el objetivo de conocer el estado del desarrollo de aplicaciones en entornos orientados a servicios.

Para realizar el diagnóstico se utilizaron varios métodos cuantitativos y cualitativos que incluyó entrevistas a profundidad con especialistas con un alto dominio en el desarrollo de aplicaciones informáticas. También se utilizaron grupos focales antes y después de aplicar el modelo y se aplicó de la técnica de IADOV para comprobar satisfacción de usuarios, así como un cuasi experimento para evaluar el comportamiento de las variables principales del modelo.

Para construir el modelo fue aplicado un procedimiento que utiliza las tendencias teóricas actuales sobre composición de aplicaciones en entornos orientados a servicios y tiene en cuenta el diagnóstico realizado para introducirlos.

La construcción conceptual del modelo exigió un diseño metodológico que abarca toda la estructura de la tesis y constituye una combinación de teoría y práctica. La vida del modelo sin embargo, lo constituye la dinámica de su funcionamiento y su aplicabilidad.

## RESULTADOS

Exposición modelo denominado MIDAC constituye el resultado fundamental de la investigación. MIDAC es una representación conceptual que integra herramientas, buenas prácticas, marcos de trabajo y una base de conocimiento de apoyo a la toma de decisiones para facilitar el desarrollo de aplicaciones compuestas.

Los principios que sustentan al modelo propuesto para la composición de aplicaciones son: la estandarización para hacer más eficiente el proceso de composición y reutilización de servicios; la interoperabilidad para lo cual es indispensable un diseño adecuado de los recursos, que permita que las aplicaciones sean compatibles y usables por cualquiera sin importar las plataformas tecnológicas donde se ejecuten proveedor y consumidor; la flexibilidad por estar basados en componentes con funcionalidades genéricas y adaptarse a las particularidades de las aplicaciones existentes para el proceso de composición; la pertinencia como garantía de la adecuación del modelo en el contexto de la orientación a servicios; la independencia funcional de los servicios lo que contribuye a su bajo acoplamiento y a su reutilización; y la actualización permanente mediante la retroalimentación de la información que nutre al modelo.

Sus principales cualidades son: la amplitud que brinda la capacidad de analizar y de emplearse en aplicaciones informáticas de múltiples dominios; el enfoque sistémico que se expresa en el modelo propuesto a través de los componentes que interactúan con vistas a perfeccionar el proceso de composición de aplicaciones; la integralidad dada por los componentes del modelo que cubren de manera integrada y coherente la mayoría de los elementos necesarios para la composición satisfactoria de aplicaciones; y la mejora continua, que se evidencia con los resultados que se van obteniendo, en particular el enriquecimiento de la base de conocimiento incorporada en el componente de selección inteligente de la información.

Las premisas con vistas a la aplicación del modelo propuesto son: la calificación de los desarrolladores condición necesaria para el uso eficiente de las herramientas propuestas para la composición de aplicaciones; la voluntad institucional que apoye la aplicación del modelo, la visibilidad de los recursos necesarios para la composición de aplicaciones y la existencia de un repositorio de servicios debidamente catalogados.

El núcleo del modelo funcional está formado por componentes relacionados entre sí, como muestra en la Figura 1:

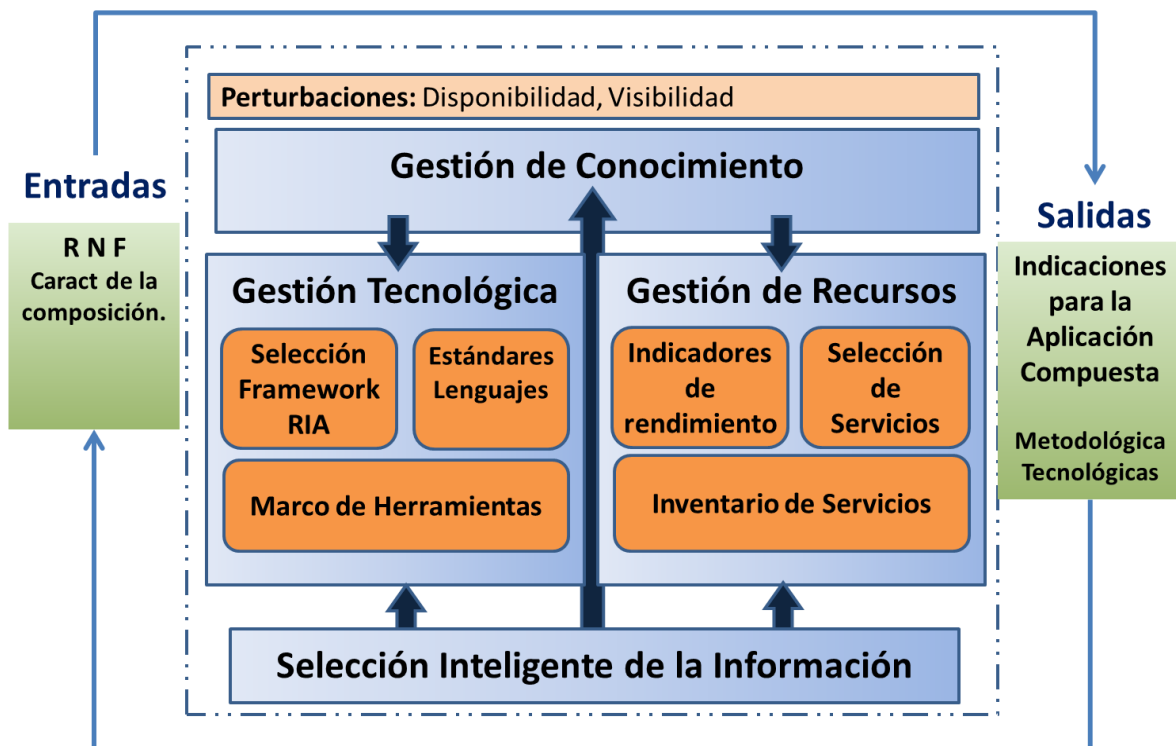


Figura 1. Representación gráfica del modelo. Fuente: elaboración propia

En el componente de Gestión Tecnológica se definen las características que deben presentar las herramientas a utilizar en el proceso de desarrollo de aplicaciones compuestas.

Este componente contiene un grupo de criterios para el desarrollo de los servicios de presentación. Específicamente apoya la selección de un marco de trabajo adecuado para la construcción de una interfaz enriquecida según las necesidades del usuario, tomando información de una base de casos estructurada y con información clasificada de los *frameworks* RIA. Esta base de casos constituye un aporte del autor así como los indicadores que la definen.

El componente Gestión de Recursos examina si los estos cumplen los requisitos de diseño. Garantiza que el almacén de servicios contiene las funcionalidades necesarias para la catalogación, el descubrimiento y la búsqueda de servicios a través de la meta información.

La Gestión de Conocimiento es un proceso fundamental en el modelo. Se gestiona tanto conocimiento tácito como explícito necesario para la creación de la base de conocimientos, la selección tecnológica y de recursos.

La selección inteligente de información se fundamenta en una base de conocimiento expresada en casos y una herramienta (SI-Holmes) para la creación de la base de casos y la realización de diagnósticos como apoyo a la toma de decisiones. Esta herramienta se construyó como parte de esta investigación. SI-Holmes es una aplicación desarrollada en el lenguaje Java y posee una interfaz amigable y configurable. Tienen la ventaja, como se muestra en la Figura 2, de interactuar al mismo tiempo con múltiples bases de casos que se encuentran almacenadas en una Base de Datos, mostrando sus nombres, cantidades de rasgos y casos de cada una de ellas.



Figura 2. Interfaz de herramienta SI-Holmes. Fuente: Elaboración propia

SI-Holmes permite crear expertos de cualquiera de sus bases de caso guardadas en Bases de Datos. Para obtener una solución de experto se selecciona el/los rasgo(s) a inferir, se llenan otros que se consideren necesarios, se escoge alguno de los algoritmos implementados para la distancia entre casos (HEOM, GOWER, ARGELIO) y el sistema indicará el resultado de la inferencia brindada por el experto y los casos más cercanos de manera gráfica y detallada. El valor fundamental del Sistema Inteligente SI-Holmes lo constituye su simplicidad y el ambiente amigable que ofrece a los usuarios. Fue desarrollado en tres versiones, una de ellas aplicando los principios del paradigma orientado a los servicios, consumiendo sus principales funcionalidades de una capa de servicios.

Se ha comprobado su funcionalidad en aplicaciones a la gestión universitaria, en el proceso de enseñanza aprendizaje y a sistemas de jurisprudencia con resultados satisfactorios.

### **Infraestructura tecnológica para la aplicación del modelo**

El ritmo creciente con el que se adopta SOA y el impacto que ha tenido ha atraído un gran número de proveedores de *software* que compiten por una cuota de mercado en la provisión de la infraestructura tecnológica requerida (Sopeña y otros, 2008). Actualmente, los proveedores de plataformas SOA se dirigen principalmente a las grandes corporaciones, por lo que el mercado está dominado por un escaso volumen de ventas y un costo asociado muy elevado.

La adopción de los modelos de desarrollo, de explotación y de negocio asociados al *software* libre representa una oportunidad para acelerar el ritmo de aparición y consolidación de estándares abiertos.

Para la infraestructura que soporte tecnológicamente a MIDAC, se seleccionaron herramientas de la suite WSO2 la cual es de código abierto y basada en estándares abiertos. Entre las herramientas más significativas que componen la suite y que el autor de esta investigación propone para la validación práctica del modelo se encuentran: el WSAS (servidor que permite almacenar aplicaciones y servicios desarrollados usando varios *frameworks*; el ESB (Enterprise Service Bus) que permite desarrollar la transparencia de ubicaciones, la transformación y el ruteo de mensajes; y el GREG (Registro/Repositorio).

### **Indicaciones metodológicas para la implantación del modelo propuesto**

A partir del análisis de los modelos de implementación propuestos en la literatura y de la experiencia en el desarrollo de soluciones informáticas en la UCI se plantean las indicaciones metodológicas para la implementación del modelo compuesto por cinco etapas o fases.

En la etapa Inicial se identifican las necesidades de la composición, su alcance, los requerimientos funcionales y no funcionales, particularidades de la aplicación que se necesita construir.

En la etapa de Selección se sugieren -empleando el componente de selección inteligente de la información- las herramientas tecnológicas adecuadas de acuerdo al tipo de aplicación que se necesita construir. La toma de decisiones se apoya en una base de conocimientos que constantemente se retroalimenta de los componentes del modelo. Se diseñan las interfaces de usuario y la lógica interna de los componentes. También se seleccionan los recursos disponibles en el repositorio de acuerdo a la funcionalidad y a los requisitos no funcionales.

En la etapa de Ensamblaje se realizan las interacciones entre los servicios proveedores dentro de las funcionalidades de la aplicación compuesta.

En la etapa de Ejecución se despliega la solución de manera que el usuario pueda acceder a los resultados de la composición y se realizan pruebas sobre ella.

Posteriormente se retroalimenta la base de conocimiento con los casos exitosos lo que constituye una actualización de la misma. Un resumen de las indicaciones metodológicas para la implantación del modelo MIDAC se presenta en la siguiente figura:

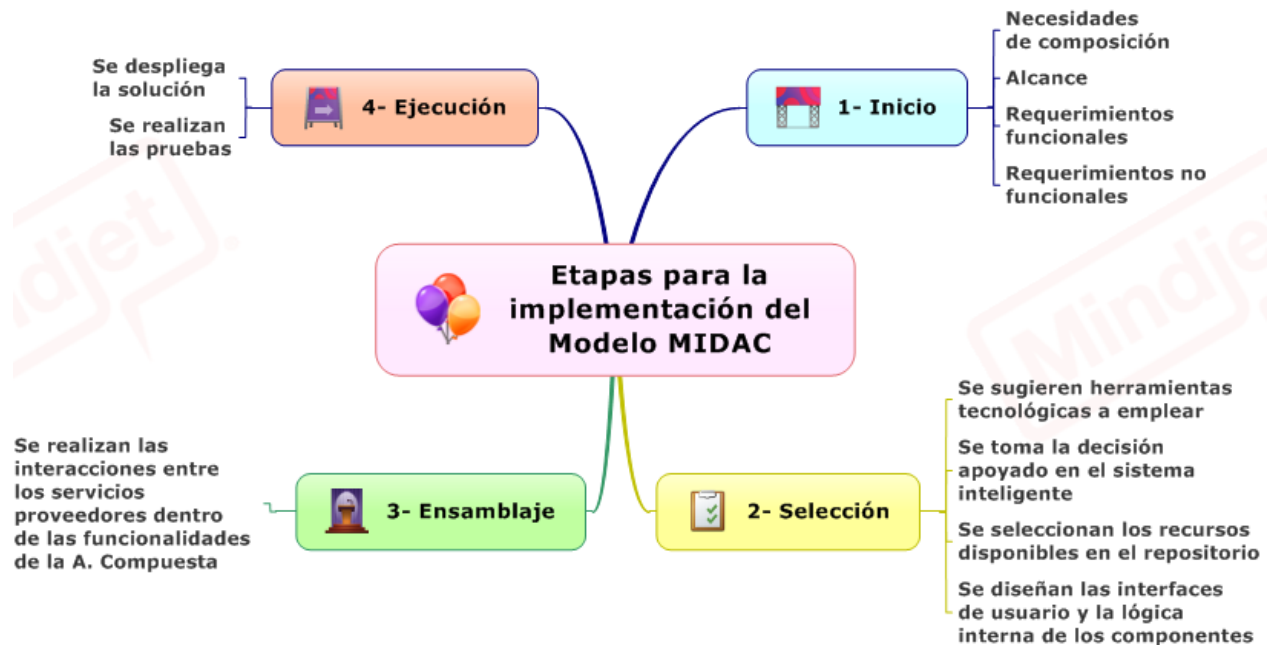


Figura 3. Etapas para la implementación del Modelo MIDAC. Fuente: Elaboración propia

## CONCLUSIÓN

- Con los elementos teóricos y prácticos más actuales de las ciencias informáticas en lo referido a las aplicaciones compuestas se obtuvo un modelo integrado que fue utilizado para la construcción aplicaciones informáticas con resultados satisfactorios y que fue validado con métodos científicos.
- El modelo garantiza que, mediante la integración de herramientas, buenas prácticas y marcos de trabajo para el desarrollo de aplicaciones compuestas en iniciativas orientadas a servicios, se puede ganar en la interoperabilidad y la estandarización del proceso.
- Se comprobó que es posible incorporar la experiencia obtenida en aplicaciones informáticas desarrolladas con anterioridad si se emplean preceptos del razonamiento basado en casos y se introducen como parte de un modelo integral. Para ello se diseñó e implementó una aplicación basada en el modelo propuesto que permite gestionar conocimiento y apoyar la toma de decisiones. De esa manera se le brinda a las organizaciones un instrumento para el registro de las mejores prácticas y el conocimiento en la gestión y desarrollo de *software*.

Para proponer el modelo MIDAC se analizaron modelos actuales para la composición de aplicaciones y se realizó un diagnóstico en instituciones productoras de *software* con el objetivo de conocer el estado del desarrollo de aplicaciones en entornos orientados a servicios.

## REFERENCIAS BIBLIOGRÁFICAS

1. Ngu, A. y otros (2010). Semantic-Based Mashup of Composite Applications. *IEEE Transactions on Services Computing*, 3(1) pp. 2-15. Recuperado de: <http://cs.adelaide.edu.au/~qsheng/papers/TSC-Ngu10.pdf>
2. Banerjee, A. (2006). What Are Composite Applications? *The Architecture Journal*, (10).
3. Bello, R. y otros (2002). *Aplicaciones de la Inteligencia Artificial*. Guadalajara: Ediciones de la Noche. ISBN 970-27-0177-5.
4. Bharadwaj, N. (2010). *BPEL PM and OSB Operational Management with Oracle Enterprise Manager 10g Grid Control*. Birmingham: Packt Publishing, ISBN 978-1-847197-74-0.
5. Blanvalet, S. y otros (2006). *Best Practices for SOA-based integration and composite applications development*. Birmingham: Packt Publishing, ISBN 1-904811-33-7.
6. Cabrera, O. y otros (2009). WeSSQoS: Un Sistema SOA para la Selección de Servicios Web según su Calidad. Recuperado de: <http://www.lsi.upc.edu/~moriol/papers/jsweb09-upc-cenidet-def.pdf>
7. Davis, J. (2009). *Open Source SOA*. Greenwick: Manning Publications, ISBN 978-1-933988-54-2.
8. Duggan, D. (2012). *Enterprise Software Architecture and Design. Entities, Services, and Resources*. Hoboken: John Wiley & Sons, Inc., ISBN 978-0-470-56545-2.
9. Dunlap, T. (2006). IBM's New Portal Server Embraces "Composite Applications". *IBM's Intranet Journal*, 24 de Agosto. Citado por: Will, R. *WebSphere SOA Highway. Portal and Business Process Management*. Recuperado de: <ftp://service.boulder.ibm.com/software/au/solutions/businessflexibility/soahighway/pdf/ManagementFINAL.pdf>
10. Erl, T. (2005). *Service-Oriented Architecture: Concepts, Technology, and Design*. Upper Saddle River: Prentice Hall, ISBN 0-13-185858-0.
11. Erl, T. (2007). *SOA Principles of Service Design*. Upper Saddle River: Prentice Hall, ISBN 9780132344821.
12. Erl, T. (2008). *SOA Design Patterns*. Upper Saddle River: Prentice Hall, ISBN 0-13-613516-1.
13. Erl, T. y otros (2010). *SOA with .NET and Windows Azure™*. Upper Saddle River: Prentice Hall, ISBN 978-0-13-158231-6
14. Febles, O. y otros (2012). Aplicación compuesta para la gestión de sistemas basados en conocimiento. *Acimed*, 23(3).
15. Febles, O. y otros (2012). Los Mashups: Aplicaciones compuestas de la Web 2.0. *Revista Ciencias de la Información*, 43(3). Recuperado de: <http://cinfo.idict.cu/index.php/cinfo/article/view/440>
16. Infante, J. (2012). UDDI o no UDDI. 2012. Recuperado de: <http://blogs.prod.uci.cu/desarrolloSOA/2010/03/uddi-o-no-uddi/>
17. Juric, M. y Krizevnik, M. (2010). *WS-BPEL 2.0 for SOA Composite Applications with Oracle SOA Suite 11g*. Upper Saddle River: Prentice Hall, ISBN 978-1-847197-94-8.
18. Matsubara, M. y otros (2009). *SOA adoption for dummies*. Hoboken: John Wiley & Sons Inc., ISBN 978-0-470-48334-3.
19. Mohebi, A. (2012). *An Efficient Qos-Based Ranking Model for Web Service Selection with Consideration of User's Requirement*. Applied Mathematics. Toronto: Ryerson University.
20. Sopeña, J. y otros (2008). Informe de Vigilancia Tecnológica 14 (VT14). *Tecnologías software orientadas a servicios*. Recuperado de: [http://www.madrimasd.org/informacionidi/biblioteca/publicacion/doc/VT/VT14\\_Tecnologias\\_software\\_orientadas\\_servicios.pdf](http://www.madrimasd.org/informacionidi/biblioteca/publicacion/doc/VT/VT14_Tecnologias_software_orientadas_servicios.pdf)