

# **PROPUESTA DE UN MODELO DE PRUEBAS PARA UNA ARQUITECTURA ORIENTADA A SERVICIOS**

**Ramón Rivero Torres, Elizabeth Ochoa Luis y Leevan Abon Cepeda**

**Centro de Consultoría y Desarrollo de Arquitecturas Empresariales (CEDAE), Universidad de las Ciencias  
Informáticas (UCI).**

## **RESUMEN**

El desarrollo de software implica una serie de actividades de producción en las que las posibilidades de que aparezca la fiabilidad humana son comunes. Debido a la imposibilidad humana de trabajar y comunicarse de forma perfecta, el desarrollo de software ha de ir acompañado de una actividad que garantice la calidad. Nuestro Centro, dado su carácter productivo, se ha insertado al igual que otras grandes empresas en el mundo al intenso proceso que existe hoy en día en Internet enfocado a los servicios Web y al estilo arquitectónico SOA, por lo que se hace necesario asegurarse de que las aplicaciones que se obtengan como resultado en esta entidad sean eficientes, seguras e íntegras.

Este modelo tiene como finalidad proponer principalmente la metodología a usar durante el desarrollo de las pruebas con el fin de ser usado por cualquier proyecto productivo que siga una línea SOA/BPM.

**Palabras clave:** modelo, prueba, servicio.

## **ABSTRACT**

Software development involves a series of production activities in which the chances that appear human reliability are common. Due to the inability human to work and communicate perfectly, the software development has to be accompanied by a quality assurance activity. Our Center, due to their production, is being inserted like other big companies in the world to the intense process that exists today on the Internet focused on Web services and the architectural style SOA, therefore it is necessary to ensure that applications obtained as result in this entity are efficient, safe and solid.

This model mainly aims to propose the methodology to be used during test development, to be used for any productive project with a line SOA / BPM.

**Keywords:** model, test, services

## **1. INTRODUCCIÓN**

Hace poco más de una década, la Arquitectura Orientada a Servicios (SOA, por sus siglas en inglés), comenzaba a escucharse por primera vez. Sin embargo, aproximadamente en el 2003, el concepto de Arquitectura Orientada a Servicios comenzó a expandirse y a encontrar sentido en un mundo en el que las aplicaciones de negocios cobran cada vez más relevancia y donde la rapidez de su puesta en marcha determina, muchas veces, el éxito o fracaso de la estrategia comercial de las compañías. SOA no es un concepto nuevo, los ingenieros de software entendieron

sus principios a mediados de los 80 cuando llegaron al mercado la computación distribuida y las llamadas a procedimientos remotos. Años después, en 1996 Gartner definió:

*“SOA es una arquitectura de software que comienza con una definición de interfaz y construye toda la topología de la aplicación como una topología de interfaces, implementaciones y llamados a interfaces. Sería mejor llamada “arquitectura orientada a interfaces”. SOA es una relación de servicios y consumidores de servicios, ambos suficientemente amplios para representar una función de negocios completa”.*

A pesar que Gartner dejó sentadas las bases de lo que sería SOA, no es hasta el 2003 que esta arquitectura entra por completo en el mundo de las TI (Tecnologías de la Información) empresariales, a través de los servicios web, surgiendo como la mejor manera de afrontar el desafío de hacer más con menos recursos, convirtiéndose en la nueva filosofía que permite soportar los requerimientos tecnológicos actuales y futuros de las empresas que estén decididas a prosperar en la nueva economía y proporcionando un puente entre TI y procesos de negocio ayudando a incrementar la flexibilidad de los mismos y la capacidad de las empresas para cumplir más rápida, fácil y económicamente las metas de negocio. Con SOA la necesidad de realizar pruebas sigue existiendo, sin embargo muchos servicios SOA pueden no tener una interfaz de usuario, lo cual es uno de los nuevos retos para la realización de pruebas al sistema.

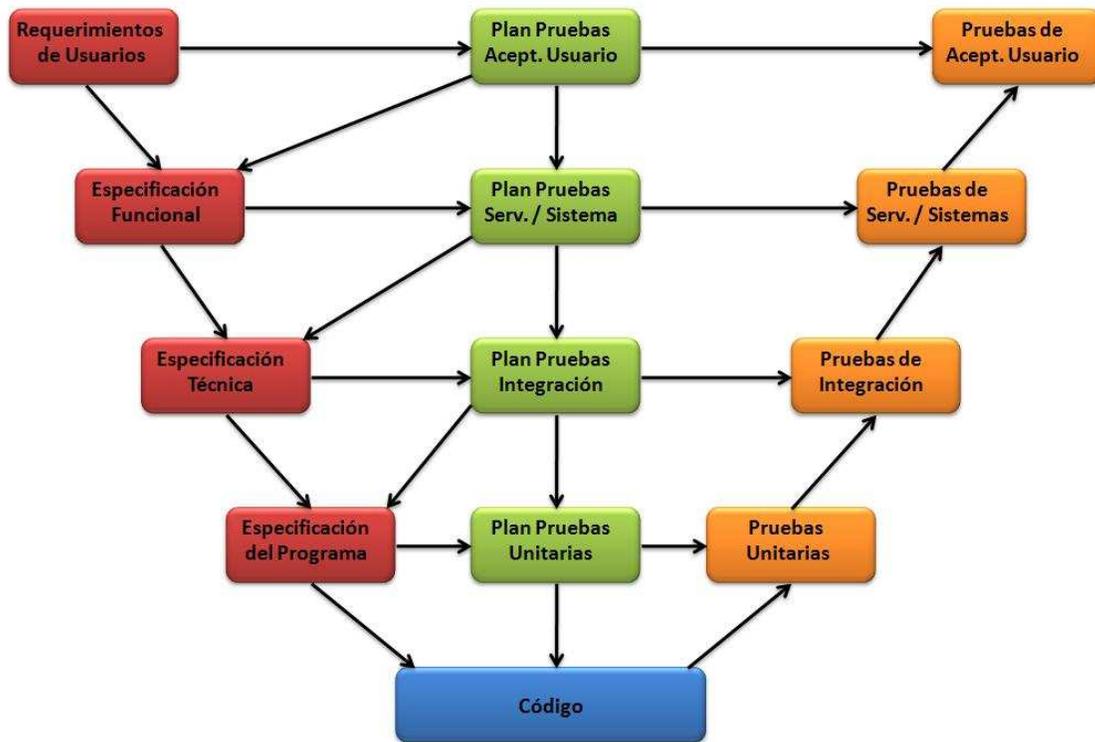
Debido a que los procesos serios de desarrollo de software, en la mayoría de los casos tienden a ser caóticos, es necesario involucrar procesos de aseguramiento de la calidad, para que se puedan cumplir de manera correcta los requerimientos que el cliente necesita. Por otro lado, el costo que implica reparar un defecto que es descubierto en etapas avanzadas del desarrollo de software, tal como la implementación, es muy alto, hablando en términos del presupuesto del proyecto, como también en el cronograma.

No probar la arquitectura SOA implica que los servicios no se hagan con la calidad necesaria, puede que no se logren conectar bien a las herramientas de procesos, arruinando toda la inversión en la iniciativa SOA de la empresa, además que los clientes se encontrarán insatisfechos con el resultado final del sistema.

## **2. DESARROLLO**

### **Descripción del Modelo V.**

El Modelo V que fuera desarrollado de manera simultánea, pero independiente, en Alemania (por el Ministerio de Defensa) y los Estados Unidos (por el Consejo Nacional de Ingeniería de Sistemas) a finales de 1980, es una evolución del modelo de desarrollo en Cascada. El modelo en cascada consiste en la ejecución secuencial de una serie de fases que se suceden, lo que da nombre al modelo. Cada fase genera documentación para la siguiente, esta documentación debe ser aprobada. Una fase no comienza hasta que la anterior ha terminado, además requiere disponer de unos requisitos completos y precisos al principio del desarrollo.



**Figura 1: Modelo V.**

La primera mitad del Modelo V es similar al Modelo en Cascada y la otra mitad tiene como finalidad hacer pruebas e integración asociado a cada una de las etapas de la mitad anterior. Esta es una de las ventajas fundamentales que presenta con respecto a otros modelos, pues involucra chequeos de cada una de las etapas, considerando las pruebas como un proceso que corre en paralelo con el análisis y el desarrollo, a lo largo de todo el ciclo de vida del proyecto, en lugar de constituir una fase aislada al final del proyecto.

También tiene la facilidad que en el momento en el cual se está realizando una fase, es posible realizar también la documentación y planeación para las pruebas que se realizarán más adelante. En la representación gráfica clásica del Modelo V, las fases de desarrollo de software aparecen a la izquierda y los correspondientes niveles de pruebas a la derecha. Partiendo de los requisitos o del diseño (a la izquierda) se deben planificar y preparar los niveles de pruebas correspondientes (a la derecha).

En general, cada actividad de pruebas a la derecha valida la actividad enfrentada de la izquierda. Además, los niveles superiores de pruebas en el diagrama están basados en caja negra (pruebas basadas en las especificaciones) y los niveles inferiores están basados en caja blanca (basadas en la estructura interna de los componentes del sistema), teniendo dichos niveles por separado, distintos objetivos, entornos y perfiles de personal.

El Modelo V es simplemente animar al equipo de proyecto a determinar continuamente cómo probar satisfactoriamente los resultados del proyecto y cada organización puede utilizar su versión, basándolo en su propia terminología. Definitivamente se trata de un modelo más robusto y completo que el Modelo en Cascada, y

puede producir software de mayor calidad. El Modelo V es un adecuado estándar de pruebas para entregar proyectos SOA por las siguientes razones:

- Estimula una metodología descendente con respecto a la definición de los requisitos del proceso de negocio, de alto nivel del diseño técnico funcional, de seguridad, etc.
- Estimula un método de prueba ascendente:
  - Prueba funciones individuales dentro de un servicio.
  - Prueba un servicio individual.
  - Prueba un conjunto compuesto de servicios a través de pruebas de un proceso integrado.
  - Prueba un sistema de negocio completo.
- En SOA los servicios son ligeramente acoplados y es por eso que un modelo de prueba ascendente es recomendable.
- Los niveles reflejan diferentes puntos de vista de las pruebas en los diferentes niveles de detalles.
- El Modelo V estimula las pruebas a lo largo de todo el ciclo de vida de desarrollo del software.

## **Estrategia de pruebas en SOA.**

### ***Propósito:***

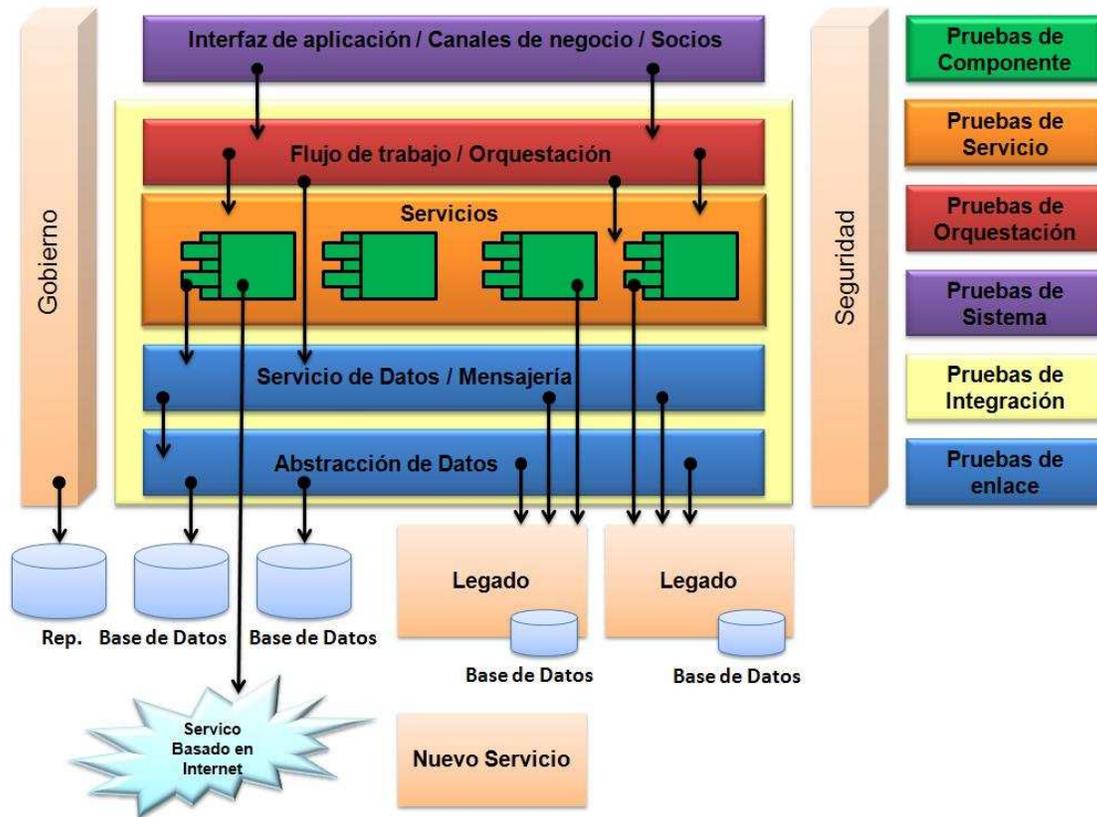
Probar una arquitectura SOA podría verse como un complejo problema de la informática. Para cualquier problema complejo, lo principal es dividirlo en pequeños problemas, en componentes más manejables y fomentar la calidad en estos. Las bases para unas pruebas satisfactorias en una arquitectura SOA son las siguientes:

- Igualdad de distribución del esfuerzo de las pruebas en todo el ciclo de vida del proyecto. Muchas organizaciones aún no reconocen los beneficios reales de las técnicas de revisión estáticas y formales durante las primeras fases del proyecto. La mayoría o la totalidad del esfuerzo en las pruebas llega demasiado tarde al final del ciclo de vida del proyecto. Se precisará de más esfuerzo en las pruebas del nivel de servicio.
- El equipo de pruebas SOA es una combinación de expertos del dominio de negocio y de la tecnología.
- Diseñar la estrategia de pruebas del proyecto junto a los requerimientos técnicos y de negocio del mismo.
- Presupuesto para el equipo de pruebas para que participen desde el inicio del proyecto.
- Implementar controles de calidad en todo el ciclo de vida del proyecto.
- Las pruebas de seguridad no son una actividad de fin del proyecto. Diseñar y planificar las pruebas de seguridad desde el inicio del proyecto.
- Las herramientas de prueba son una necesidad.

### ***¿Cómo se prueba una arquitectura SOA?***

¿Cómo se prueba una arquitectura SOA? Pues no existe una definición estándar para hacerlo. En cambio podemos aprender a descomponer la arquitectura en sus componentes, trabajando desde los más simples hasta los más complejos, probando cada uno de ellos. En otras palabras, se tiene que dividir la arquitectura en dominios como los servicios, seguridad y gobierno y probar cada dominio por separado usando la estrategia y herramientas definidas por los indicadores propuestos.

SOA está formada por servicios de aplicación débilmente acoplados y altamente interoperables y un enfoque de pruebas en SOA debe seguir el mismo patrón.



**Figura 2: Componentes de pruebas aplicados a SOA.**

La figura anterior representa un modelo de los componentes de pruebas aplicados a SOA y como estos están interrelacionados. El equipo de pruebas diseña la estrategia de pruebas del proyecto y los planes deben tener un macro entendimiento de cómo todos los componentes trabajan tanto independiente como colectivamente.

### Fases o niveles de pruebas y tipos de pruebas.

Se definen las pruebas en SOA en las siguientes fases o niveles:

- Nivel de prueba de Componentes de Servicios.
- Nivel de prueba de Servicios.
- Nivel de prueba de Integración.
- Nivel de prueba de Proceso/Orquestación.
- Nivel de prueba de Sistema.

#### *Nivel de prueba de Componentes de Servicios.*

En el nivel de pruebas de Componentes de Servicios, las pruebas de Unidad son normalmente realizadas por los desarrolladores para probar que el código no solo compile satisfactoriamente, sino que la funcionalidad básica de los componentes y funciones dentro de un servicio estén trabajando como lo especificado. El objetivo primario de las pruebas de Componentes de Servicios es tomar pequeños pedazos de software probable en la aplicación, aislarlo del código restante, y determinar si el comportamiento es justamente como se esperaba. Cada componente

es probado por separado antes de integrarlo dentro de un servicio o servicios. Las siguientes actividades de calidad y prueba son recomendadas en esta fase/nivel de prueba:

- Revisiones formales de código para asegurarse que cumple con los estándares de la organización y para identificar cualquier función potencial y defectos de seguridad o debilidad.
- Los criterios de calidad de entrada y salida no son solo definidos para este nivel de prueba, sino que son alcanzados antes de moverse al siguiente nivel de pruebas.

### ***Nivel de prueba de Servicios.***

Las pruebas de Servicios constituyen la fase/nivel más importante dentro de la Metodología de Pruebas en una SOA. Las siguientes actividades de calidad y prueba son recomendadas en esta fase/nivel de prueba:

- Revisiones formales de código para asegurarse que cumple con los estándares de la organización y para identificar cualquier función potencial y defectos de seguridad o debilidad.
- Pruebas funcionales, de rendimiento y seguridad son ejecutadas contra los servicios. Esto exigirá la ayuda de herramientas de pruebas automatizadas.
- Los criterios de calidad de entrada y salida no son solo definidos para este nivel de prueba, sino que son alcanzados antes de moverse al siguiente nivel de pruebas.

Las pruebas del nivel de servicio deben garantizar que el servicio no solo cumpla los requisitos del proyecto actual, sino que es más importante todavía el cumplimiento de los requerimientos operacionales y de negocio de los otros procesos que están usando ese servicio.

### ***Nivel de prueba de Integración.***

La fase de pruebas de Integración se enfocará en las interfaces de los servicios. Esta fase de prueba apunta a determinar si el comportamiento de la interfaz y la información compartida entre los servicios, está trabajando como lo especificado. El equipo de prueba asegurará que todos los servicios entregados a esta fase de pruebas cumplen con lo definido en la definición de la interfaz, en términos de normas, la validación del formato y los datos. Los escenarios de prueba de las pruebas de Integración también deberían trabajar las capas de comunicaciones y los protocolos de red.

### ***Nivel de prueba de Proceso/Orquestación.***

Las pruebas de Proceso/Orquestación aseguran que los servicios estén funcionando colectivamente como lo especificado. Esta fase de pruebas cubriría la lógica del negocio, la ordenación en secuencia, manipulación de excepciones y descomposición de procesos (incluyendo rechazo de servicios y procesos).

### ***Nivel de prueba de Sistema.***

Las pruebas del Nivel de Sistema formarán la mayoría, si no todas las Fases de Prueba de Aceptación de Usuario. Esta fase probará que la solución técnica de la arquitectura SOA ha dado los requerimientos del negocio definido y ha encontrado los criterios de aceptación del negocio definido.

La siguiente tabla muestra las fases o niveles de pruebas y los tipos de pruebas que se realizan en cada fase o nivel.

**Tabla 1: Niveles de pruebas / Tipos de pruebas**

Fases de prueba	Documentos guía	Funcionales	Rendimiento	Interoperabilidad	Compatibilidad con Versiones Anteriores	Aceptación	Seguridad
Nivel de Componente	Esquema Técnico, Especificaciones y Normas de Aplicación	✓	✓	✓	✓	✓	✓
Nivel de Servicio	Requerimientos del Negocio, Esquema Técnico & Políticas y Normas de Gobierno	✓	✓	-	-	✓	✓
Integración & Orquestación	Requerimientos del Negocio, Esquema Técnico & Políticas y Normas de Gobierno	✓	-	✓	✓	-	✓
Sistema/Proceso (Aceptación de usuario)	Requerimientos del Negocio, Esquema Técnico & Políticas y Normas de Gobierno	✓	✓	✓	✓	✓	✓

***Pruebas de Gobierno.***

El Gobierno SOA es un factor fundamental en el éxito de cualquier implementación SOA. Es también el término más “débilmente” usado, ya que abarca todo el ciclo de vida de la implementación SOA, desde el diseño hasta el tiempo de ejecución en curso de mantenimiento. El Gobierno SOA se refiere a los Estándares y Políticas que gobiernen el diseño, construcción y puesta en práctica de una solución SOA y las políticas que deben aplicarse durante el tiempo de ejecución.

Las organizaciones deben tener bien definido el diseño, el desarrollo, las pruebas y las normas de seguridad que guiarán y dirigirán implementaciones SOA. Los controles de calidad y revisiones deben llevarse a cabo a través de todo el ciclo de vida del proyecto y los procesos, para garantizar el cumplimiento. La contraparte adecuada debe conducir esas revisiones y los cambios con respecto a los parámetros estándar de las normas recomendadas que deben ser acordadas por el equipo de gobierno de la organización. Los siguientes son ejemplos de tipos de políticas del gobierno SOA:

- Políticas de calidad de los servicios en rendimiento, seguridad y transacciones.
- Políticas de regulación.
- Políticas de negocio (reglas).
- Políticas de auditoría (qué eventos necesitan ser registrados, cuánto tiempo se debe mantener un evento).
- Políticas de infraestructura (acceso, copias de respaldo, recuperación de desastres y rescate de la conexión).

Los casos de prueba serán construidos y ejecutados en todas las fases de prueba del proyecto para determinar si las políticas SOA se están aplicando. Las políticas SOA pueden ser aplicadas en tiempo de ejecución, mediante el uso de tecnologías y/o herramientas de monitoreo. Las pruebas del gobierno SOA no estarán en una fase de prueba separada. Las pruebas que el gobierno SOA aplicará tomarán lugar en todo el ciclo de vida del proyecto, a través de revisiones homólogas y diferentes escenarios de prueba que serán ejecutados durante las fases de pruebas separadas.

### ***Pruebas de Seguridad.***

Como SOA evoluciona y crece dentro de su organización, el perfil y la necesidad de pruebas de seguridad aumentarán. Hoy, varias organizaciones realizan una inadecuada cantidad de pruebas de penetración al final de un proyecto. SOA en combinación con el gobierno y el cumplimiento normativo, requerirá actividades de pruebas de seguridad que serán incorporadas dentro del ciclo de vida completo del proyecto.

Muchos procesos de negocio dentro de una organización se componen de varios servicios, físicamente ubicados en diferentes partes de la red corporativa, actualizando una serie de bases de datos y datos potencialmente sensibles compartidos con organizaciones externas. Esto hace que nos planteemos la pregunta de “¿Qué tan segura es la información ya que navega en una compleja red tanto interna como externa?”.

Hoy muchas organizaciones realizan pruebas de penetración de la seguridad al final del ciclo de vida del proyecto para cubrir toda la seguridad del software. Las pruebas de penetración son un ensayo autorizado de violación de la seguridad de un sistema usando un agente externo y/o técnico de acceso gusano. Realizando pruebas de penetración en el final de un proyecto corre un riesgo significativo no solo de encontrar graves errores de seguridad sino también de la entrega de un sistema que tiene un inadecuado diseño de seguridad.

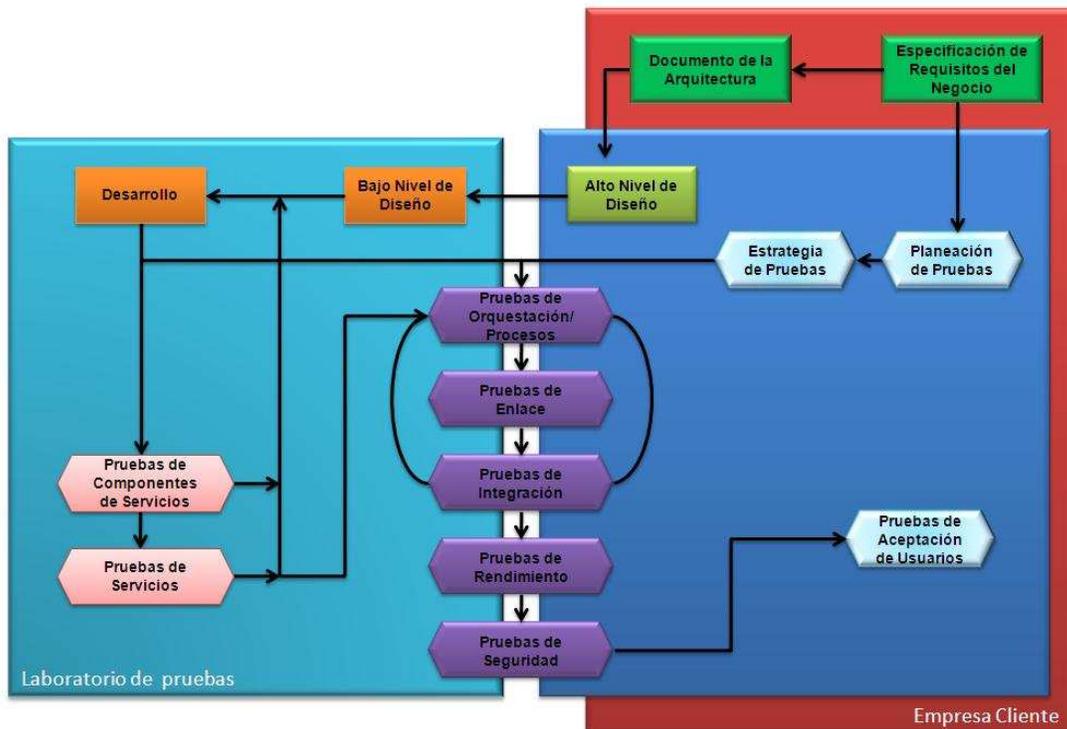
Como SOA evoluciona, las redes de las organizaciones llegarán a ser más complejas y no será posible proteger a todos los activos. La prioridad de seguridad será requerida para proteger los activos más valiosos de la compañía.

- La definición de requerimientos del negocio debe incluir requerimientos de seguridad.
- Una evaluación de riesgos de seguridad debe realizarse durante la fase de diseño técnico para priorizar y justificar las pruebas de seguridad requeridas.
- Revisar formalmente todos los resultados técnicos que se han construidos de acuerdo a las normas de seguridad definidas por sus organizaciones.
- Las pruebas de penetración de seguridad pueden ser planificadas y ejecutadas en el *nivel de componentes de servicios* y no solo cuando ha sido entregado un sistema totalmente integrado.

Hoy, hay disponibles muchas herramientas de pruebas de seguridades tanto comerciales como libres. Estas herramientas han evolucionado de dispositivos de exploración que informan de los posibles puntos débiles de seguridad a las herramientas que realmente ejecutan específicos tipos de pruebas de penetración. Las herramientas de seguridad son necesarias si su organización desea un acreditado y repetible método de pruebas de seguridad.

### **Modelo de Pruebas Integrado.**

Este modelo ofrece soluciones reales a los nuevos desafíos de las pruebas en SOA, precisando significantes esfuerzos y actividades de prueba en el nivel de servicio. La principal razón detrás de dicha afirmación es que los servicios serán reutilizados. Si un servicio tiene defectos conocidos así como otras cuestiones de calidad, entonces probablemente no será seleccionado para ser reutilizado por el equipo de desarrollo. SOA exigirá que los servicios individuales sean entregados a las fases de pruebas de integración y aceptación de usuario con las declaraciones y las garantías de calidad en la funcionalidad, rendimiento y seguridad del negocio.



**Figura 3: Modelo de Pruebas Integrado.**

La figura anterior representa una vista general del proceso de desarrollo de las pruebas de calidad en una arquitectura orientada a servicios. El modelo se encuentra dividido en dos partes, una parte que tendría lugar en la empresa cliente y la otra en el laboratorio de pruebas. Existen actividades como la especificación de los requisitos, la planeación y estrategia de pruebas, así como las pruebas de aceptación de usuarios (últimas a realizar durante el proceso de desarrollo) que son obligatorias realizarlas en el área del cliente. En cambio, parte del diseño, el mayor peso del desarrollo del sistema, las pruebas de componentes de servicios y las pruebas de servicios, son actividades realizadas en el laboratorio de pruebas. Existen también un conjunto de pruebas que pueden ser llevadas a cabo en ambos lugares, como son el caso de las pruebas de orquestación, de enlace, de integración, de rendimiento y de seguridad.

### 3. AUTORIZACION Y RENUNCIA

Los autores autorizan a LACCEI para publicar el escrito en los procedimientos de la conferencia. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que esta expresado en el escrito.

### 4. CONCLUSIONES

El presente trabajo finaliza definiéndose un modelo flexible y acorde a los requisitos necesarios para la realización de pruebas de calidad y lo más importante es que es un modelo que puede ser utilizado por cualquier empresa o proyecto de desarrollo que siga una línea SOA/BPM. El mismo está compuesto por la metodología a usar durante

el desarrollo de las pruebas, los entregables que recogerán todos los datos de cada fase, los roles involucrados y además se definieron indicadores para la selección de herramientas de pruebas.