

WHITE PAPER

Los desafíos del desarrollo de Aplicaciones Ricas para Internet (RIA) y cómo superarlos:

Con una tecnología de Plataforma de Aplicaciones de punta a punta.

Por Avigdor Luttinger,
VP Estrategia Corporativa

Abril de 2009

Magic Software es marca registrada de Magic Software Enterprises Ltd. y todo otro producto o nombre de empresa mencionado es propiedad de sus respectivos propietarios. Magic Software Enterprises hace su mejor esfuerzo para asegurar que la información aquí expresada es correcta; sin embargo, no se ofrece garantía sobre esta información. Información sujeta a cambios sin previo aviso y no representa un compromiso de Magic Software Enterprises o sus representantes. Magic Software is a trademark of Magic Software Enterprises Ltd. All other product and company names mentioned herein are for identification purposes only and are the property of, and may be trademarks of, their respective owners. Magic Software Enterprises has made every effort to ensure that the information contained in this document is accurate; however, there are no representations or warranties regarding this information, including warranties of merchantability or fitness for a particular purpose. Magic Software Enterprises assumes no responsibility for errors or omissions that may occur in this document. The information in this document is subject to change without prior notice and does not represent a commitment by Magic Software Enterprises or its representatives.



Introducción

Los valiosos beneficios que las aplicaciones Ricas de Internet (RIA) representan para las empresas ya han sido ampliamente documentados y son claramente reconocidos; desde un bajo costo total de propiedad, hasta la disponibilidad total de la aplicación, mayor seguridad y una experiencia del usuario mejorada.

Claro que, mientras un RIA Empresarial provee ventajas medibles (vitales para mantener la competitividad), estos beneficios tienen también un precio: la complejidad del sistema.

Este *white paper* explora los desafíos del desarrollo de Aplicaciones Ricas de Internet efectivas y ofrece un mapa de ruta para un diseño de RIA que esquivе los problemas actuales y maximice los potenciales beneficios.

Hasta hace poco, habían sólo dos caminos hacia la distribución efectiva de RIA completas:

- 1. Usar una Plataforma-como-servicio (PaaS).** Con ejemplos como Force.com, Google, Microsoft, Amazon y LongJump. Si esta fuera la opción elegida, entonces deberá considerar que si bien todas las partes del sistema son provistas por usted, la plataforma está todavía en algún lugar en la "Nube".
- 2. Usar una plataforma de de tipo "cliente":** Para empresas con *peros* acerca del concepto que involucra ubicar sus activos en la "Nube", la única forma de adquirir capacidades para construir RIA era hasta hace poco, hacerlo por su propia cuenta empleando alguna de la variedad de herramientas de tipo "Plataforma de Cliente RIA", como Adobe Flex/Air, Java FX o Microsoft Silverlight.

Con RIA usted obtiene:

Menor costo total de propiedad.

Con RIA no hay necesidad de instalar software del lado del Cliente ni ocuparse de mantener a cada nuevo usuario. La aplicación se configura automática y transparentemente en el lado del Cliente y toda la aplicación y las actualizaciones de la misma y así como de la plataforma se llevan a cabo en el servidor .

Estar listo para SaaS

Las aplicaciones SaaS representarán el 25% de todo negocio de software de aquí al 2011*. RIA es un componente esencial en el desarrollo de una propuesta exitosa de Software-como-Servicio (SaaS).

Mejor experiencia del usuario y productividad

A diferencia de las aplicaciones de Internet típicas, las RIA no requieren una comunicación Cliente-Servidor para cada dato procesado y cada actividad. Su aplicación corre más rápido y más agradablemente, sus empleados hacen su trabajo más rápido y sus clientes reciben un mejor servicio.

Más disponibilidad de la aplicación.

Las RIA pueden ejecutarse remotamente desde cualquier parte y pueden accederse en cualquier momento vía un amplio abanico de dispositivos móviles portátiles. Sus empleados pueden acceder a los datos necesarios para trabajar desde su propia casa o donde sea fuera de la empresa.

Mejor seguridad corporativa y protección de derechos de propiedad intelectual

La arquitectura multi capa de RIA le permite ocultar los elementos sensibles de su aplicación y evitar el robo corporativo, el abuso del activo "clientes" y el *malware*.

(* Gartner Report: SaaS Delivery Challenges On-Premise Software, 26th September, 2006)

Detrás del brillo del momento: La realidad de la “Nube”.

Para pequeños negocios o aquellos en crecimiento, la “Nube” computacional conlleva considerables ventajas. Nuevas capacidades de TI como almacenamiento y procesamiento pueden agregarse incrementalmente en vez de incurrir en los costos totales asociados a la instalación de servidores y su mantenimiento. Los servidores también ocupan lugar: requieren *datacenters*, administradores, aire acondicionado, suministro de electricidad ininterrumpida, resguardo de datos, seguridad, actualización de versiones, optimización de performance, parches y reemplazo de partes de hardware, cuando fallan. La “Nube” informática ofrece una atractiva y simple forma de tercerizar esta pesada carga como retorno de una sencilla suscripción de bajas barreras de entrada y salida para aquellos negocios nuevos en la arena. Por supuesto, una vez tomada la decisión de transferir sus sistemas en línea, moviendo el almacenamiento local y el procesamiento al modelo de la computación de la *Nube*, habrá efectivamente tercerizado sus datos de misión crítica a una empresa externa. Más allá del brillo mediático del momento, la *Nube* consiste en servidores, discos duros y conexiones de red. La *Nube* es real. *El Cyber Espacio* es un lugar, y por lo tanto es susceptible, igualmente que el

hardware local lo es, a desgaste, detención y servicios de baja calidad.

A medida que dependemos más de la Nube, la vulnerabilidad también se incrementa, y en forma directa con el desafío permanente de la protección de datos. También es importante recordar que los proveedores son compañías, como cualquier otra, y se pueden equivocar. ¿Que pasará cuando las planillas de cálculo, los datos de los clientes y los presentaciones de ventas resulten repentinamente inaccesibles por un problema cualquiera del lado de la plataforma del proveedor?.

- ***Vulnerabilidad a la recesión***

Mientras los efectos de esta recesión continúan sintiéndose en todo sector de nuestra economía, vemos ya sin sorpresa que hasta los más grandes y estables nombres de empresas pueden archivarse por bancarrota y desaparecer de la noche a la mañana; a veces sin aviso a sus clientes los que quedan solo para juntar los pedazos . ¿Que pasaría si mañana éste fuera el caso del proveedor de su plataforma?.

La respuesta es, en consecuencia, mantener la opción de operar los sistemas de TI de misión crítica internamente, siempre y cuando su costo lo hiciera factible.

Pero aquí nuevamente, las empresas se encuentran enredadas en la falta de eficiencia en el rango de las plataformas para la construcción del RIA de tipo del lado “Cliente”.

La falacia de las soluciones de sólo lado “Cliente”.

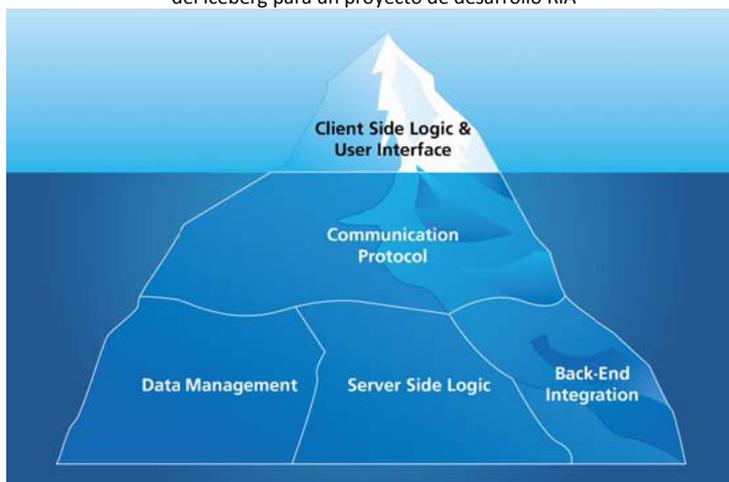
Mientras los responsables de TI pueden por supuesto sentirse atraídos por el concepto de *Enterprise RIA* y desean experimentar y probar con sus propias manos el concepto en sus organizaciones, ellos habrán primero de entender las implicaciones de construir su propia plataforma “Cliente” RIA. Más allá de crear una linda interfase, lo que puede considerarse como la punta del *iceberg*, hay mucho más por debajo de la superficie de RIA que considerar.

Debajo de la superficie “cliente”, se esconden procesos complejos de desarrollo que implican mezclar delicadas partes en *permanente movimiento*, las que pueden convertirse en el mayor obstáculo al desarrollo, y que podrían terminar con las expectativas, forzando a los CIO’s a archivar los proyectos y abandonar los beneficios prometidos. Serían así malas noticias para departamentos de TI en la búsqueda de mejorar sus servicios y dar a la empresa el soporte que necesita para competir en el presente clima de negocios.

El Iceberg RIA. El mayor desafío de desarrollo.

Las aplicaciones ricas de Internet representan uno de los más desafiantes procesos de desarrollo. Involucran tres capas: una capa Cliente (responsable de la presentación, la lógica de interacción y algo de la lógica del negocio), la capa Servidor (responsable de la mayor parte de la lógica del negocio y la integración del *back-end*), y la capa de la Sesión (la capa de comunicación entre el cliente y el servidor, la que requiere ciertas capacidades de programación de sistemas bastante particulares).

Desarrollar el lado Cliente y la interfase con el usuario representa apenas la punta del Iceberg para un proyecto de desarrollo RIA



Entonces, normalmente, el esfuerzo de desarrollo RIA requiere del armado y mantenimiento de distintos equipos trabajando en diferentes aspectos de la aplicación. Como resultado, el diseño, la planificación y la administración del proyecto se vuelven más riesgosos, y por supuesto, más caros. Como todo sistema, cuanto más partes en movimiento, más chances de fallas a lo largo de la línea.

- ***El desafío del acoplamiento.***

Las aplicaciones Cliente-Servidor tradicionales involucran una arquitectura relativamente sencilla, que descansa sobre una permanente conexión entre el Servidor y el Cliente. Con tal diseño *fuertemente acoplado*, casi no se requiere manejar explícitamente ni preservar distintos estados de lógica. Por el contrario, las aplicaciones web, que centralizan su procesamiento en el Servidor, dejan al Cliente esencialmente *desacoplado, o débilmente acoplado*. Mientras las aplicaciones web realicen pequeños y simples procesos lógicos, y tengan una limitada riqueza de interactividad, pueden implementarse con arquitecturas web Standard y un simple manejo de sesión.

La arquitectura RIA procura por supuesto proveer el confort de la experiencia del usuario altamente interactiva y *fuertemente acoplada*, pero con una configuración *débilmente acoplada*. Esto requiere un sofisticado diseño de sistemas y capacidades de programación que raramente se encuentran en equipos de desarrollo focalizados en soluciones de negocios. Actualmente las únicas plataformas que soportan este tipo de arquitectura sin tener que recurrir a programación de nivel de sistema son las llamadas plataformas **basadas en metadatos**, como es el caso de Magic Software uniPaaS, Force.com y Cordys.

- ***Superación de las limitaciones del ancho de banda.***

Con el incremento sostenido del ancho de banda, más y más recursos están disponibles para las aplicaciones web, llevándonos, con tecnologías como Ajax, cerca del nivel del tipo de interacción del Cliente-Servidor o de una LAN. Esto se puede apreciar en la función “*Google Suggest*” que trae un rango de términos de las búsquedas más populares a medida que uno ingresa las letras en el campo de búsqueda. Esta funcionalidad puede ser apropiada para tareas como la navegación de páginas web, pero el ancho de banda actual de Internet no es suficiente para soportar aplicaciones *fuertemente acopladas*, con decenas de campos activos por pantalla.

Para superar esta limitación, una aplicación de tipo RIA debe soportar más lógica en el lado del Cliente y también particionar el procesamiento entre el Servidor y el Cliente. Por lo tanto los desarrolladores terminan trabajando con dos procesos físicamente separados y lógicamente dependientes, que además se ejecutan en tandem en ambos Cliente y Servidor. El cliente ahora precisa de un “cerebro” y “memoria” para encarar las constantes secuencias de instrucciones del Servidor. Mantener una sesión con coherencia requiere en consecuencia una administración sofisticada del estado y sesión.

Mientras las aplicaciones Cliente Servidor tradicionales requerían solo un desarrollador orientado al negocio, para crear una plataforma de tipo *sólo* Cliente una aplicación de tipo RIA, a ejecutarse en la propia instalación (es decir, no alojada en servidores bajo el control de terceros), se requiere de incorporar un programador de sistemas al equipo, agregando un monto considerable al costo total de la solución.

La muralla del desarrollo

Como las plataformas RIA de tipo Cliente están hoy día ampliamente difundidas entre los desarrolladores web, vemos que algunas compañías comienzan con iniciativas en RIA concebidas en el departamento de diseño web. Por naturaleza, los desarrolladores web tienden a focalizarse en el enriquecimiento de la experiencia que atañe a la interfase con el usuario de la aplicación; y para ello, con las tecnologías como *Flex* o *JavaFX*, ciertamente lo consiguen. Sin embargo, frecuentemente los desarrolladores web no cuentan con una vista de Sistemas panorámica, y por ello pueden encontrarse con dificultades en cuanto a arquitectura apenas avanzan más allá de la capa Cliente. Algunos intentan utilizar *Ajax*, pero esto tiene también limitaciones de performance y en muchos casos el proyecto llega hasta ese nivel de avance. Es muy común encontrar también que un equipo de desarrollo web dedique medio año para desarrollar una interfase para el

usuario del tipo *sólo* Cliente. Luego se encuentran con una muralla infranqueable: mientras las herramientas para construir el Cliente proveerán el “cerebro” y “memoria” del cliente para poder superar las limitaciones de ancho de banda de la web, por otra parte no proveerán la conectividad ni la lógica del negocio, capacidades que no se encuentran muy a menudo en estos equipos de trabajo. Inclusive, aquellas organizaciones que perseveran desarrollando las tres capas del proceso completas encuentran luego dificultades al tener que mantener la solución actualizada y en consonancia con las tecnologías y estándares en permanente evolución. Esto es un gran problema que las organizaciones que están experimentando con RIA recién están comenzando a descubrir.

Performance y optimización

De lo visto hasta aquí, resulta claro que desarrollar y desplegar una aplicación de tipo RIA involucra múltiples escenarios de comunicación entre las partes *Cliente* y *Servidor* de la aplicación.

Al suscribirse a una plataforma de punta a punta alojada en servidores bajo el control de un proveedor, éste se ocupa de realizar las inversiones en los recursos necesarios para asegurar que la aplicación se encuentre ajustada y optimizada para rendir al máximo.

Por supuesto, cuando una organización elige usar alguna herramienta de desarrollo de tipo Cliente como las mencionadas,

ésta tendrá que desarrollar sus capacidades de ajuste de performance internamente.

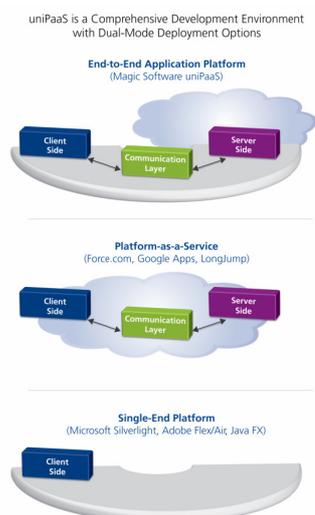
Como tal tarea requiere las capacidades especializadas con las que muchas organizaciones de negocios no están generalmente familiarizadas, se verá la empresa nuevamente forzada a invertir más tiempo y recursos para ajustar su aplicación al nivel de una performance y optimización comparativamente inferior a la de una alternativa alojada por un proveedor.

El ajuste fino de performance involucra la reducción de los escenarios de comunicación entre el Servidor y el Cliente al mínimo posible y la reducción del tamaño de los paquetes a fin de limitar el uso de la red y así alcanzar el tiempo óptimo de respuesta de la aplicación. El uso de herramientas exclusivamente del tipo *lado Cliente*, como las antes mencionadas, significa que el desarrollo del lado Servidor y del Cliente se llevan a cabo con diferentes paradigmas y equipos de trabajo, lo que resulta en una mayor posibilidad de mala utilización del ancho de banda.

Otro problema es que esta falta de optimización solo sale a la luz luego de que ha sido realizado el desarrollo de la aplicación completa, y recién cuando los clientes y usuarios comienzan a reclamar sobre la pobre performance. Los desarrolladores tendrán entonces que volver a re-optimizar los escenarios de comunicación, incurriendo en mayores costos y pérdidas de tiempo; que no resultan en otra cosa ganancia que en la de una mala reputación ante los clientes involucrados en el proceso.

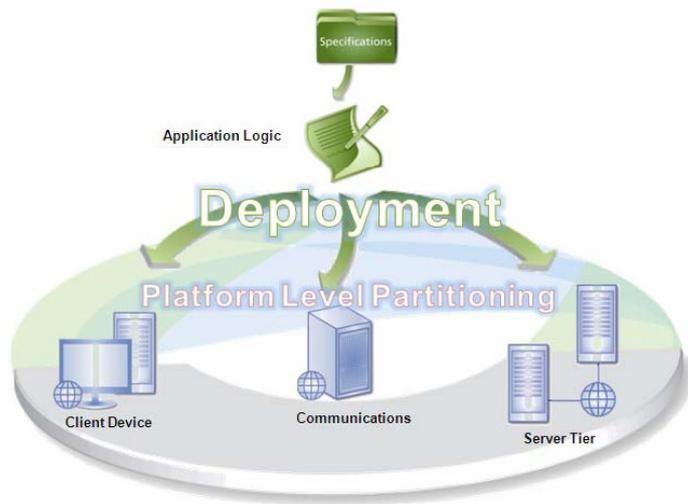
La solución “punta a punta”

Una nueva generación de plataformas RIA punta a punta ya está en el mercado y pueden proveer una respuesta abarcativa a los desafíos de desarrollo y ejecución de RIA. **Una de éstas plataformas es Magic Software uniPaaS, que le brinda a los usuarios la libertad de elegir la forma de ejecutar la aplicación, tanto localmente, en su instalación, como en la “Nube”, sea como plataforma alojada u ofrecida en forma de Plataforma-como-Servicio (PaaS).**



La diferencia clave es que este tipo de plataforma “completa” provee todas las partes de la solución; incluyendo la *porción hundida del iceberg*, sin requerir un desarrollo separado. De aquí la denominación de “punta a punta”. **Esta solución utiliza metadatos en vez de código “hard-code”, de forma que lo único que realmente se ha de hacer es describir la lógica de negocios de la aplicación y diseñar la interfase de usuario requerida.** La plataforma se encarga del resto, en forma similar a las capacidades de una solución PaaS alojada por terceros, **pero sin las limitaciones de la ejecución de la Nube.**

uniPaaS particiona y despliega automáticamente la Lógica de la aplicación entre el componente de comunicación de la aplicación que opera tanto en el Cliente como en el Servidor.



Empleando un paradigma unificado de desarrollo, uniPaaS puede entregar aplicaciones web RIA y SaaS incorporando todos los aspectos del desarrollo y despliegue de procesos bajo un único techo.

Mediante la combinación de ambas lógicas de los lados Cliente y Servidor dentro de un mismo paradigma, los negocios pueden evitar los costos y esfuerzos de reunir los diferentes recursos separados que se requieren para construir aplicaciones web RIA, reduciendo en consecuencia el riesgo del proyecto y eliminando una gran parte del costo de mantenimiento del mismo una vez que la aplicación se encuentra en plena producción.

uniPaaS maneja toda la configuración y control del lado del Cliente y del Servidor, además la comunicación Cliente-Servidor, incluyendo el consumo y manipulación de servicios del *back-end*; todo empleando el mismo lenguaje.

uniPaaS es también la única solución que soporta todo el espectro de formas de distribución de aplicaciones (desktop, client/server, web, RIA y SaaS), con la misma copia de la aplicación.

<i>Plataforma de Aplicaciones uniPaaS</i>	<i>Herramientas RIA más difundidas</i>
<i>Plataforma de punta a punta</i>	<i>Plataforma para el lado cliente exclusivamente.</i>
<i>Metadatos (codificación opcional)</i>	<i>Programación intensiva</i>
<i>Actualización de Aplicaciones manejada por la plataforma de manera transparente.</i>	<i>Requiere codificación explícita de la Actualización de la Aplicaciones vía API.</i>
<i>Administración de la sesión y el contexto.</i>	<i>Requiere codificación explícita del manejo de la sesión y el contexto.</i>
<i>Un único conjunto de capacidades se requieren para manejar tanto el front end como el back end</i>	<i>Requiere tres conjuntos de capacidades para desarrollar un proyecto completo.</i>
<i>Soporte de la plataforma .NET</i>	<i>Soporte de Cliente multiplataforma.</i>

Conclusión

Nunca fue tan fácil como ahora, contando con una plataforma punta a punta basada en metadatos, navegar el mar de desafíos que significa construir aplicaciones empresariales RIA; y la recompensa nunca fue tan provechosa: una experiencia del usuario que sobrepasa la más *enriquecida* aplicación cliente servidor, más el increíble hecho de que esté disponible a través de una conexión *http* asegurando que empleados y asociados de negocio se beneficien de aplicaciones totalmente funcionales en el momento en que ellos lo requieran. En definitiva, se cuenta hoy con un escenario *win-win* para las empresas proveedoras de soluciones de TI y sus clientes.

Para más información sobre Magic Software y la plataforma de aplicaciones uniPaaS visite www.magicsoftware.com.ar