

Aplicaciones tradicionales, un activo para subir a la nube

Las estrategias y metodologías más apropiadas para crear, desarrollar y operar aplicaciones en la nube están entre las principales preocupaciones de sectores de la academia y de la industria relacionados con el *cloud*. Algunos de sus representantes se encontraron en el 10.º Foro de *Cloud Computing*, “DevOps: Desarrollo, despliegue y operación de aplicaciones en la nube”, organizado por el Departamento de Ingeniería de Sistemas y Computación (DISC) y que tuvo lugar el 2 de junio del 2016.



Durante la instalación del 10.º Foro de Cloud Computing, Harold Castro presentó la investigación desarrollada por el grupo Comit (Comunicaciones y Tecnología de Información) del DISC y Heinsohn.

Foto: Natalia Fernanda Madrid Vidales

El control y la automatización de cada etapa de los procesos de desarrollo y operación de las aplicaciones para una mayor disponibilidad y seguridad en la nube son las características de la cultura que proponen DevOps (nombre compuesto a partir de las palabras *Development* y *Operations*) y SRE (*Site Reliability Engineering*). Sobre sus bondades en seguridad habló el invitado especial de este Foro Isis, Camil Samaha, *Solutions Architecture Senior Manager* en Amazon Web Services (AWS) (pág. 40). Por su parte, Andrés Villarroel, de Globant, expuso la cultura SRE (pág. 42) nacida en Google. Los otros cuatro participantes, Santiago Gil y Carlos Valencia, de Heinsohn, junto con Harold Castro y Mario Villamizar, del DISC, presentaron resultados de la investigación “Construcción de una línea de productos de aplicaciones de crédito/cartera que serán ofrecidas a través de un *marketplace* en forma SaaS para el sector no bancarizado (pymes) de Colombia”, dirigida por Harold Castro Barrera, realizada por el grupo de investigación Comit (Comunicaciones y Tecnología de Información) del DISC y Heinsohn, con el apoyo de Colciencias. En este artículo se resumen sus intervenciones.

Modernizar no es rehacer

Subir a la nube aplicaciones construidas hace 15 años, con tecnología de esa época,

“Infraestructura como servicio (IaaS) permite moverse entre proveedores, ofrece auto escalabilidad y control de los servicios, aunque en lo relacionado a utilizar servicios gestionados implica atar la aplicación a algunos servicios”.

Mario Villamizar

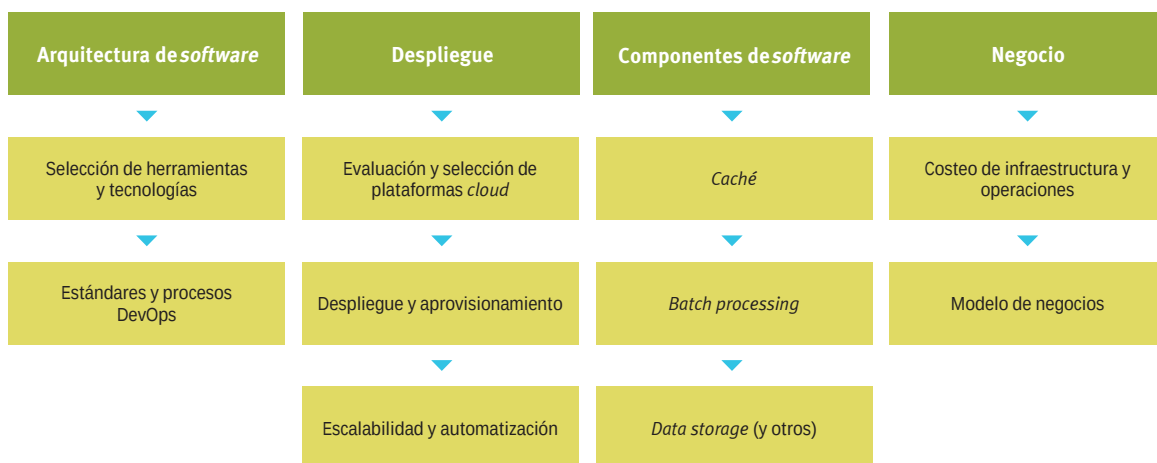
para alcanzar mayor eficiencia, con beneficios como la autoescalabilidad y costos de operación más bajos era el reto del estudio. Este surgió de los requerimientos habituales de los clientes de Heinsohn que no quieren perder esos activos, por lo que no están conformes con la propuesta de los proveedores de rehacer tales aplicaciones.

El profesor Castro introdujo la charla afirmando que para tomar ventaja de los beneficios de la nube hay que entender lo que está disponible y destacó la necesidad de que los procesos administrativos relacionados con estas investigaciones sean más rápidos, pues a pesar de que Colciencias aprobó el proyecto en el 2012, “solo pudimos empezar en el 2014. No podemos ser competitivos si vamos mucho más lento que la industria”.

También dijo que aunque como ingenieros de sistemas les preocupa el desarrollo de la aplicación, es indispensable

entender que detrás de ella está el modelo de negocio de Heinsohn, que señala cómo debe ser la entrega de la operación de una solución a un cliente. “La nube nos movió todo este escenario. Ya no se trata de vender la licencia, sino el servicio; el desarrollo no es para la infraestructura del cliente, sino para una que escalará en unas dimensiones distintas; y ya no se desarrollará una aplicación, que puede evolucionar de manera diferente para cada cliente, sino que el mismo *software* se debe poder personalizar, con el fin de aprovechar las economías de escala. Esa problemática ya no es aterradora como lo era en el 2012, pero estamos convencidos de que todavía hoy muchas compañías y proveedores de soluciones la siguen enfrentando”.

Explicó que, una vez logrado el desarrollo, hay que ponerlo a funcionar, pero no con un despliegue tradicional, sino en un tiempo determinado, operando para



Estructura de la investigación y desarrollo de metodologías y tecnologías cloud, enfocadas en las soluciones del sector financiero de Heinsohn Business Technology.

muchos clientes a una escala determinada. Y destacó: “Esto cambia el tratamiento que se le da a la seguridad, pues hay que incluirla desde el principio del proceso y monitorearla de manera constante. Estas condiciones transformaron la forma de trabajar”.

¿Lo requiere el negocio?

El director del DISC le dio paso a Santiago Gil, gerente del área de Arquitectura de Heinsohn, quien contó que el desafío de llevar a la nube una aplicación antigua se hizo en dos frentes: en los productos y servicios proveídos por su empresa y en los productos y servicios del cliente, en los que Heinsohn cumple las funciones del área tecnológica. “Las empresas quieren aprovechar los beneficios del *cloud* porque en las aplicaciones antiguas no hay autoescalabilidad y los costos de operación, de consumo de *hardware*, de comunicaciones son altos, pues varios ingenieros se deben encargar de esas tareas, lo cual redundan en la oferta comercial”. Sin embargo, aclaró, conviene establecer los requerimientos de negocio para llegar a la nube, pues la inversión no vale la pena si no es en beneficio del mercado y del usuario.

Afirmó que los principales problemas de las aplicaciones antiguas están asociados con la usabilidad y los múltiples dispositivos, asuntos técnicos que se resuel-

Carlos Valencia y Santiago Gil, de Heinsohn, y Mario Villamizar, profesor del DISC, hablaron de las distintas etapas y componentes de la investigación conjunta que ha migrado a la nube aplicaciones tradicionales, desarrolladas por esta compañía colombiana de diseño de *software*.

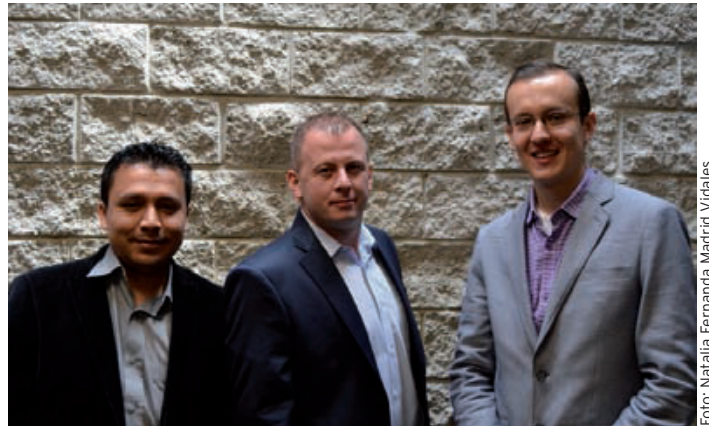


Foto: Natalia Fernanda Madrid Vidales

ven en la capa de presentación —algo así como el 50 % de una aplicación—, de tal forma que hay que rehacerla.

Cómo elegir un proveedor

Por su parte, Mario Villamizar presentó las alternativas evaluadas para cumplir el requerimiento. Las opciones eran: 1, dejar las aplicaciones sin ningún cambio y moverlas a una solución de infraestructura como servicio; 2, analizar cuál es el esfuerzo de mover una aplicación antigua a una solución de infraestructura como servicio para aprovechar los beneficios de la infraestructura, en particular el autoescalamiento; 3, escoger una solución de plataforma como servicio para no gestionar toda la infraestructura, ya que, en gran parte, esto lo hace el proveedor. 4, el

software como servicio no era viable porque se quería desplegar una aplicación ya desarrollada.

En la primera alternativa —la más simple— no son necesarios los cambios en desarrollo, se conserva la arquitectura de la aplicación con las tecnologías de presentación, pero la desventaja es operativa y de costos, que se elevan porque no hay autoescalabilidad. Solo se puede escalar de forma estática: hay que detener los servidores para incrementar la carga de cómputo. Este modo de operación no es eficiente en costos.

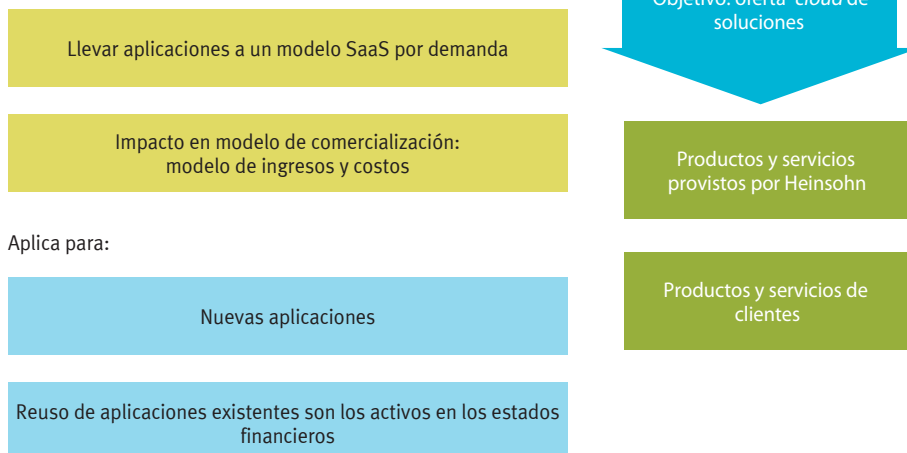
La solución de plataforma como servicio (PaaS) ofrece variedad de opciones y el proveedor se encarga de gestionar gran parte de la infraestructura. Esto funciona muy bien cuando la aplicación se acopla al *stack* (los componentes) tecnológico para el cual el proveedor diseñó su servicio de la nube. En el caso de estudio, la aplicación estaba desarrollada sobre el de JEE (*Java Platform Enterprise Edition*).

Sin embargo, como estas plataformas no lo soportan, el análisis se centró en la validación de los requerimientos de las aplicaciones para ser desplegadas sobre un proveedor tipo IaaS, utilizando mecanismos de autoescalamiento.

También se examinaron los motivadores, entre los cuales está la facilidad de moverse de un proveedor a otro, pues en un PaaS hay cierta dependencia.

Se seleccionó desplegar la aplicación sobre una solución de infraestructura como servicio (IaaS), pues permite moverse entre proveedores, ofrece autoescalabilidad

Motivadores de negocio



Motivadores de negocio para hacer la migración de aplicaciones de Heinsohn.

y control, aunque implica atarse a algunos servicios y se requieren cambios en ciertos despliegues si se va de uno a otro.

El profesor Villamizar hizo hincapié en la necesidad de verificar si el escogido todavía ofrece servicios gestionados listos para utilizar, además de considerar la maduración del producto y que la oferta evoluciona constantemente. Y dijo: “Mi aplicación utiliza caché, base de datos, colas: ¿debo montarlos sobre los servicios básicos de máquinas virtuales? ¿O el proveedor ya los tiene como servicios gestionados y solo hay que pagar sin preocuparse de cómo estos escalan?”.

Luego, utilizando una matriz, se evaluaron diversos aspectos de esas compañías, como asuntos financieros, oferta de servicios, certificaciones técnicas y de seguridad; si en infraestructura solo soportan máquinas virtuales, servicios de almacenamiento, bases de datos, balanceos de carga, colas y sistemas de caché.

Con la gestión de operación se hizo lo mismo para estudiar el tipo de servicios que entrega cada operador. Uno de los temas al que se prestó mayor atención fueron los mecanismos de autoescalamiento con el fin de establecer si facilitan el crecimiento de la capacidad computacional de forma automática, aunque para muchos proveedores ya es un estándar. Otros suministran las API (*Application Programming Interface*) y el cliente es quien hace los desarrollos según sus políticas. También se analizó cómo es el monitoreo de los servicios y otros factores no técnicos como los de privacidad.

De este examen detallado, dijo el profesor Villamizar, se concluyó que lo más importante es que la compañía elegida y quien la contrata tengan alineados de la misma manera los componentes y requerimientos técnicos y de negocio.

El proveedor no lo hace todo: hay que adaptar

Carlos Valencia, arquitecto de *software* de Heinsohn habló de detalles técnicos que enfrentaron en el proceso de migrar a la nube y dijo que en la mayoría de los casos se trató de mantener la arquitectura.

“Aunque se seleccionó AWS como proveedor, algunos de sus servicios no resultaron apropiados para las necesidades, por lo que nos apoyamos en programas de otras compañías”.

Mario Villamizar

Uno de los requisitos importantes fue el de conservar la transaccionalidad en el manejo de sesiones, pues cada vez que se crea una nueva instancia se corre el riesgo de perder algo. En las aplicaciones antiguas “teníamos muchas dependencias de las librerías usadas, muchos procesos y funcionalidades como auditorías, históricos, problemas de *login* por el contenedor de aplicaciones utilizado”.

Aunque se seleccionó AWS como proveedor, algunos de sus servicios no resultaron apropiados para las necesidades, por lo que se apoyaron en programas de otras empresas. Por ejemplo, “el grupo de *autoscaling* y el balanceador de Amazon eliminaban instancias al desescalar, pero no funcionaban bien ya que se perdían las sesiones en las instancias eliminadas. Para salvar las sesiones, buscamos balanceadores externos que permitieran distribuir las sesiones en las diferentes instancias sin pérdida de información. HAProxy (*software* de código abierto) nos funcionó muy bien”.

Según Carlos Valencia, las métricas por defecto de AWS no son las más elocuentes en términos de negocio. “Necesitábamos saber cómo se comporta la aplicación con bastante carga, pues ante este tipo de periodos la aplicación en JEE experimenta un mayor consumo de memoria, métrica que no mide por defecto AWS. Entonces habilitamos la ejecución de un agente en las instancias Linux para conocer cuánta memoria gasta la aplicación”. Las métricas en Amazon CloudWatch permiten visualizar el comportamiento de las instancias que se despliegan y en particular las métricas personalizadas de uso de memoria.

En el proceso de análisis se presentaron inquietudes que Carlos Valencia compartió: “¿Qué pasará cuando no tengamos mucha carga? Las sesiones de cada instancia generada al escalar, se perderán al momento de desescalar. ¿Cómo solucionarlo? Apagando los equipos que están menos utilizados. ¿Cómo? El DNR (*Dynamic Network Routing*) debe recibir sesiones para las instancias menos utilizadas”.

Luego, se optimizó el despliegue para que se demore solo 30 segundos, pues si hay muchas peticiones se requiere crear un servidor dinámicamente. Este proceso no se puede tardar ocho minutos, al que debe sumársele lo que se demora Amazon en generar un nuevo equipo y en desplegar una nueva instancia. “Ocho minutos para encender una instancia es mucho tiempo y es posible que cuando esté disponible ya no se necesite”.

Para establecer si se debe escalar o no se utilizó el servicio de notificaciones de Amazon, que permite poner en cola los eventos. De esta forma, el balanceador se enteró de cuándo balancear o distribuir las peticiones entre las nuevas instancias.

El expositor coincidió con sus compañeros de proyecto al afirmar que la adopción del *cloud* no es trivial y destacó en las lecciones aprendidas lo siguiente:

- Cada cambio requiere ajustar la aplicación a ciertas características del proveedor de manera diferente. “Los cambios de arquitectura son muy costosos. Los que nosotros hicimos son pequeños”.
- Conviene revisar la selección de IaaS versus PaaS de acuerdo con los requerimientos.
- La migración a la nube debe responder a motivadores de negocio reales. ■