

SG[®]

SOFTWARE GURU

NO.50

CONOCIMIENTO EN PRÁCTICA
www.sg.com.mx

APACHE
STREAMING
PAG. 18

SPARK

USER
DESIGN
PAG. 34

EXPERIENCE

PRUEBA DE SOFTWARE
PAG. 40

FUNDAMENTOS

ENTENDIENDO
MAPREDUCE

PAG. 50

REPORTAJE

INDUSTRIA DE
SOFTWARE
EN COLOMBIA

PAG. 12

LA EMPRESA DIRIGIDA POR DATOS

¡Que no lo sepa tu jefe!

El 80% de tu inversión anual en capacitación se va a la basura.



¿Lo evitarás este año? ¿O esperas que tu jefe lo haga?

¡Nosotros te ayudamos!

- La transferencia del aprendizaje comienza al terminar cada curso.
- El retorno sobre tu inversión en capacitación no llega al terminar el curso.
- Más del 90% de las empresas en México no hacen nada al respecto.



Nuestro compromiso contigo no termina con un curso, termina con el éxito de tus proyectos.

+52 (55) 5594 6411

cursos@abiztar.com.mx



Pregunta por los **ambientes de aprendizaje de Abiztar** y nuestras tecnologías de aprendizaje.



Pregunta por nuestra **garantía universal de aprendizaje**, única en la industria de capacitación en México.

VISITA abiztar.com.mx/video





¡Actualízate
con los gurús!

WEBINARS GRATUITOS

CURSOS PAGADOS

CURSOS GRATUITOS

Más de 5,000 miembros

Más de 12 mil horas de
capacitación online impartidas



 SGCampus

 @SGCampus

www.sgcampus.com.mx

MÁS DE LA MITAD DE LOS EX ALUMNOS TRABAJAN EN PUESTOS DE DIRECCIÓN

SOLO HAY UN **ITAM**



INGENIERÍAS

Telecomunicaciones
Computación
Mecatrónica
Industrial
Negocios

POSGRADOS

Maestría en Ciencia de Datos
Maestría en Ciencias en Computación
Maestría en Tecnologías de Información y
Administración

Contamos con diversas alternativas para obtener becas y ayuda financiera según el programa de tu elección.

Tel: (55) 5628 4028, 01 800 000 ITAM
informes@itam.mx, www.quieroseritam.com.mx
f Quiero ser ITAM t Quieroser_ITAM
posgrados@itam.mx, www.posgrados.itam.mx
f Posgrados ITAM t PosgradosITAM

SG

VIRTUAL CONFERENCE

10a. edición

Construyamos mejor software sin fronteras

Más de 20 conferencias virtuales sobre herramientas, tecnologías y mejores prácticas para el desarrollo de software, impartidas por ponentes internacionales



27 de abril 2016
EVENTO GRATUITO
Sede: Todo el mundo

¡Regístrate ahora!
sg.com.mx/sgvirtual

Mayores informes
eventos@sg.com.mx
Tel: +52(55) 5239-5502

 SGSoftwareGuru

 @RevistaSG

SG[®]

SOFTWARE GURU

NO.50 | CONOCIMIENTO EN PRÁCTICA
www.sg.com.mx

EN PORTADA

LA EMPRESA DIRIGIDA POR DATOS

020

¿Cómo es que las empresas pueden obtener ventajas competitivas por medio de la analítica de datos?

I INDUSTRIA Y EMPRESAS

Los Emprendedores de TIC en México 008

T HERRAMIENTAS Y TECNOLOGÍAS

Radar 016

Panorama 018

V VOCES

En Defensa del Software 048

El Software Orientado al Consumidor 049

P PRÁCTICAS

Resolviendo Dilemas en un Pipeline de Data Science 030

Cómo Iniciar Tu Carrera en UX Design 034

¿Sigues Siendo Importante el Aseguramiento de Calidad? 036

Resistencia al Cambio y Enfoque en el Cliente 038

F FUNDAMENTOS

Entendiendo MapReduce 050

C COLUMNAS

Tejiendo Nuestra Red 006

Tendencias en Software 017

Prueba de Software 040

Programar es un Modo de Vida 042

Clave Morsa 044

R REPORTAJE

Industria de Software en Colombia 012

Retrospectiva 054

O EN CADA NÚMERO

Noticias y eventos 005

Hardware 052

Humor 053

Biblioteca 056



Viejos los cerros



● ... y todavía reverdecen.

Cincuenta números se dicen fácil, pero han involucrado más de 10 años y más de mil artículos.

Todo esto ha sido posible gracias a los cientos de colaboradores que han escrito en SG. Agradecemos a todas y cada una de las personas que a lo largo de los años han dedicado tiempo a compartir conocimiento con sus colegas por medio de nuestras páginas. En especial estamos muy agradecidos con nuestros columnistas, quienes en cada número nos inspiran a seguir aprendiendo y aplicando.

El tema principal de este número es la analítica predictiva de datos. Es una tendencia que está impactando todo tipo de negocios y es clave que conozcamos sus bases y las oportunidades que representa.

Al igual que los 49 números anteriores de SG (50 si contamos el 0), este ha sido preparado con gran cariño y dedicación. Esperamos que lo disfrutes y te sea útil.

El equipo de Software Guru

SG es posible gracias a la colaboración de

Dirección Editorial **Pedro Galván** | Dirección de Operaciones **Mara Ruvalcaba** | Dirección Comercial **Claudia Perea**
Coordinación Editorial **Susana Tamayo** | Arte y Diseño **Oscar Sámano** | Suscripciones **Mariana Torres**
Consejo Editorial: Luis Daniel Soto | Gunnar Wolf | Luis Vinicio León | Hanna Oktaba
Ariel Jatuff | Emilio Osorio | Gloria Quintanilla | Jorge Valdés

COLABORADORES EN ESTA EDICIÓN

Alejandra Jiménez, Manuel del Pino, Jorge Gálvez, Miguel Nhuch, Blanca Vargas, Andrés Arteaga, Eduardo Flores,
Misael León, Roselyn Piñango, Masa Maeda, Manuel López Michelone.

EQUIPO SG

Coordinación de servicio **Yoloxochitl Juárez** | Marketing y Vinculación **Wendy Fabela**
Developer Relations **Luis Sánchez e Hilda Ramírez** | SG Campus **Ariel García**
Contacto: info@sg.com.mx

SG Software Guru es una publicación trimestral editada por Brainworx, S.A. de C.V., San Francisco 238 Altos. Col. Del Valle. Los contenidos de esta publicación son propiedad intelectual de los autores y se hacen disponibles bajo licencia Creative Commons Attribution-NonCommercial 4.0 International. Todos los artículos son responsabilidad de sus propios autores y no necesariamente reflejan el punto de vista de la editorial.

Reserva de Derechos al Uso Exclusivo: En trámite. ISSN: 1870-0888. Registro Postal: PPI5-5106. Distribuido por Sepomex.

Noticias

CTO FORUM

1

Software Guru realizó una reunión más de su comunidad de CTOs con el objetivo de conocer tendencias, compartir ideas y hacer networking. La reunión se llevó a cabo en el recién inaugurado Centro de Innovación BBVA en Ciudad de México. El tema que dominó la agenda del día fue la gestión de datos, con ejecutivos de Tableau compartiendo su visión sobre las 10 tendencias para 2016 en inteligencia de datos, y Delphix presentando una propuesta para habilitar datos en aplicativos por medio de Data as a Service. Mantente al tanto de la próxima reunión en <http://meetup.com/ctos-mx>



FINNOSUMMIT BOGOTÁ 2016

2

La capital de Colombia fue sede de Finnosummit, el evento de referencia para el sector de tecnología financiera en Sudamérica. El evento brindó la oportunidad de analizar los avances tecnológicos más disruptivos en torno a la industria financiera y al mismo tiempo. Además se complementó con FINNOSUMMIT Challenge competencia regional que eligió a las dos startups Fintech más prometedoras de Sudamérica, resultando ganadoras: Aflore una red de consejeros informales que distribuyen productos financieros a través de una plataforma tecnológica y Alegra una aplicación en la nube que permite a pequeñas empresas gestionar sus gastos y facturas desde cualquier lugar.

HACKMTY

3

HackMty se realizó el 20 y 21 de febrero en las instalaciones del Tecnológico de Monterrey Campus Monterrey con la participación de 400 hackers y el apoyo de empresas como Intel Software, Icalia Labs, Segundamano, y Cemex entre otras. En su segundo año, HackMty se ha convertido en el hackathon estudiantil más grande en México, atrayendo estudiantes de todo el país e incluso a algunos internacionales. Conoce la lista de ganadores en <http://hackmty.com>



MICROSOFT CLOUD ROADSHOW EN MÉXICO

5

TSP SYMPOSIUM MÉXICO 2016

4

El 23 y 24 de febrero se llevó a cabo el TSP Symposium México 2016, organizado por el Software Engineering Institute y el SIE Center de México, con apoyo de la Secretaría de Economía a través de su fondo ProSoft 3.0. En él se dieron cita más de 270 personas para escuchar keynotes, participar en talleres y actualizarse en temas importantes para la industria como la calidad de software, seguridad y arquitectura de software, TSP SM y nuevas tecnologías e innovación y nuevos negocios.

Sao Paulo y Ciudad de México fueron las ciudades en Latinoamérica elegidas para unirse al Microsoft Cloud Roadshow, un evento itinerante realizado en 15 distintas ciudades del mundo. La empresa aprovechó el evento para difundir ante la comunidad tecnológica la variedad de su oferta de servicios en la nube, así como refrendar su compromiso de empoderar a personas y organizaciones a hacer más a través de su nube inteligente. Brad Anderson, Vicepresidente corporativo de Enterprise Client and Mobility comentó que las empresas en México están acelerando rápidamente su adopción de la nube y que estiman que para 2018 el 90% de las organizaciones mexicanas utilicen al menos una solución hospedada en Microsoft Azure.



Propósito 2016: Organicémonos un Poco Más

Por Hanna Oktaba



La Dra. Hanna Oktaba es profesora de la UNAM y su objetivo principal es generar conocimiento a través de la creación y promoción de estándares. @hannaoktaba

● **Les quiero comentar** sobre algunas iniciativas de organización de la comunidad de la Computación y, en particular, de Ingeniería de Software en las cuales me tocó participar como miembro fundador en los últimos años.

RED UNIVERSITARIA DE COLABORACIÓN DE INGENIERÍA DE SOFTWARE Y BASE DE DATOS (REDIS&BD)

En mayo del 2006 se constituyó en la UNAM la Red Universitaria de Colaboración en Ingeniería de Software y Base de Datos (RedIs&BD), coordinada por Guadalupe Ibarguengoitia González de la Facultad de Ciencias. Su misión es integrar a un grupo de universitarios expertos para conformar una red de colaboración, intercambio y difusión del desarrollo y aplicación de software de calidad, para beneficio de la Institución. Durante 10 años la actividad de la red se ha concentrado en la organización de seminarios mensuales y talleres, así como encuentros universitarios; el último organizado en octubre de 2015, logró conjuntar interesados no solo de la UNAM sino de otras instituciones académicas y privadas. Más información en <http://www.redisybd.unam.mx>

RED TEMÁTICA MEXICANA DE LA INGENIERÍA DE SOFTWARE (REDMIS)

Es un proyecto surgido de la Convocatoria de Registro y Estructuración de Redes Temáticas CONACYT 2014. Su objetivo es integrar una red de especialistas en Ingeniería de Software, agrupando a los investigadores y académicos así como a los empresarios con intereses comunes en la disciplina, con el fin de generar conocimiento científico, generar buenas prácticas de manera colegiada, elaborar y ejecutar propuestas que apoyen la educación en el área, así como unir esfuerzos con la industria para apoyar su competitividad en el ramo del desarrollo de software. El responsable técnico de REDMIS es Reyes Juárez Ramírez de la Universidad Autónoma de Baja California y en su Consejo Técnico se encuentran académicos de la UNAM, UPSLP, UAM-I, UTM, ITH, UADY y UV. La comunidad está organizada por nodos ubicados en diversos lugares del territorio mexicano. La principal actividad de la REDMIS es dar continuidad a las reuniones académicas de Ingeniería de Software realizadas desde el año 1997. La próxima es la reunión de CONISOFT16 que se llevará a cabo en abril en la UPAEP, Puebla (<http://redmis2016.com.mx>). Más información en <http://redmis.com.mx>

ACADEMIA MEXICANA DE COMPUTACIÓN (AMEXCOMP)

Tiene como antecedente directo a la Red Mexicana de

Investigación y Desarrollo en Computación (REMIDEC) fundada en 2008 para hacer un análisis de la computación desde su introducción a México, su estado actual y sus perspectivas, en el marco de la conmemoración de los 50 años de la computación en México. El Consejo de REMIDEC, integrado por los doctores Luis A. Pineda (IIMAS UNAM), Christian Lemaitre (UAM Cuajimalpa), Carlos Coello (CINVESTAV), Enrique Sucar (INAOE), Jesús Favela (CICESE) y Francisco Cantú (ITESM), planteó la posibilidad de crear la Academia Mexicana de Computación como una Asociación Civil. La constitución formal de la Academia Mexicana de Computación AC sucedió el 9 de enero de 2015 con 19 miembros fundadores (entre ellos su servidora como la única mujer). El objetivo principal de la Amexcomp es integrar a los investigadores, ingenieros, tecnólogos, innovadores, docentes y promotores de la computación más distinguidos de México en las distintas áreas de la computación para constituir un foro de excelencia de pensamiento computacional que sirva como referencia central de la ciencia y la tecnología computacional en México. En su primera asamblea general en diciembre de 2015 se crearon las secciones académicas por áreas de computación. Una de ellas es la de Ingeniería de Software de la cual soy coordinadora. Más información en <http://amexcomp.mx>

Como pueden ver hay un interés de la comunidad para organizarse con diferentes objetivos y alcances. Los lectores de esta columna están cordialmente invitados a unirse a ellas. Sin embargo, me parece que nos hace falta mucho esfuerzo de lograr la cohesión y colaboración para que la sociedad, la industria y el gobierno nos conozcan, aprecien y aprovechen. Tenemos que hacer más "ruido". SG es uno de los pocos ejemplos en este sentido, que hay que apoyar y seguir.

Al principio de 2016 lancé por twitter una llamada "Que hagamos algo sensato para la industria de software en 2016! ¿Qué se les ocurre?". Recibí una sola propuesta pero, en mi opinión muy buena, "Concretar la constitución de una comunidad de profesionales que fomente la generación de ideas aplicables a la realidad mexicana" de Alexandra Alatorre (@alex_alatorre). Sobre todo me gustó lo de "ideas aplicables a la realidad mexicana". Creo que tenemos una fuente inagotable de oportunidades de innovación en serio, a través de la solución con TI de los problemas que padecemos. Pero la mayoría sueña con competir con Facebook o exportar a E.U.

Para la discusión constructiva y propuestas los invito a @hanna.oktaba. 🌐



KMMX

CAPACITACIÓN EN TI, WEB Y MOBILE

Usabilidad Mobile



Duración: 20 hrs.
Inicio: 28 de marzo

NativeScript



Duración: 30 hrs.
Inicio: 2 de abril

Programación con R



Duración: 40 hrs.
Inicio: 2 de abril

IoT con Omega Onion



Duración: 15 hrs.
Inicio: 4 de abril

Desarrollo para iOS



Duración: 30 hrs.
Inicio: 28 de marzo

LPIC-1

Linux Professional
Institute Certification



Duración: 30 hrs.
Inicio: 2 de abril

www.kmmx.mx

 @KMMXmx

 facebook.com/KMMXmx

Sucursal Roma

San Luis Potosí 211, piso 8,
Col. Roma, D.F. CP 06760

52645422 •

Sucursal Condesa

Campeche 300, piso 1,
Col. Condesa, D.F. CP 06100

68191094



Los Emprendedores de TIC en México

● **El emprendimiento en las tecnologías de la información** es quizá el aspecto más disruptivo del mundo en las últimas décadas. Hoy las nuevas empresas de tecnologías de la información están acaparando la atención por su alto crecimiento. Son las únicas, a lo largo de la historia bursátil, que en menos de 5 años han adquirido un valor mayor al del total de empresas que cotizan en las bolsas de distintas economías. Tal es el caso de Whatsapp que se vendió por 19 mil millones de dólares con apenas 52 empleados y con menos de 5 años de antigüedad. Este dinamismo explica por qué la inversión de capital en Estados Unidos en los últimos veinte años creció 5 veces más rápido en el sector TIC que en el resto de la economía.

Es así que en 2014, el Instituto Mexicano para la Competitividad (IMCO) con el apoyo de Microsoft, realizó un estudio para conocer las principales barreras y recomendaciones para impulsar el ecosistema de emprendimiento de TI en México. Para ello entrevistó a más de 300 emprendedores y expertos de la industria.

Este artículo presenta un resumen de los principales hallazgos y recomendaciones.

La versión completa del reporte está disponible en http://imco.org.mx/banner_es/los-emprendedores-de-tic-en-mexico

INNOVACIÓN

Este es el factor más rezagado de acuerdo a los emprendedores encuestados por el IMCO. Esto se debe principalmente a la baja inversión en investigación y desarrollo (I+D) en comparación con otros países. Mientras que en México 30% de la inversión en I+D

proviene del sector privado, en otros países como Corea del Sur, ésta representa 80%. La mayor parte del dinero en I+D es público y se concentra principalmente en la investigación académica con un monto 2.5 veces mayor al de la investigación tecnológica. Por si fuera poco, prácticamente no hay vínculos institucionales entre el sector académico y la industria; en otras palabras las empresas no recurren a la academia para innovar, lo que impacta de manera negativa. Un estudio reciente encontró que tan solo el profesorado y los estudiantes de la Universidad de Stanford habían creado 40,000 empresas y generado 5.4 millones de empleos desde 1930. Pero más importante es que dichas empresas generaron un valor agregado de 2.7 trillones de dólares, es decir mayor al tamaño de la economía mexicana. De acuerdo al Reporte Financiero de la Universidad de Stanford de 1995, el gobierno federal representaba 40% del financiamiento para la institución en la materia.

Otra forma de verlo, es a través de la brecha en ingresos que se generó entre Corea y México desde 1980. Esta brecha de prácticamente el doble se explica en parte a causa de la innovación entre los dos países. Mientras que el pago por regalías en Corea es de 144 dólares por habitante en México es de 6.5 dólares.

Para resolver los retos de este factor a partir de política pública, IMCO propone, entre otros:

- Modificar los incentivos del Sistema Nacional de Investigadores.
- Crear un mecanismo para dar viabilidad comercial a patentes tecnológicas.
- Crear centros de vinculación entre los sectores público, privado y académico.

ENTORNO PARA HACER NEGOCIOS

México está en el lugar 65 dentro del informe Doing Business 2016 del Banco Mundial en la categoría de apertura de empresas con mayor facilidad. De acuerdo con el reporte, las economías donde es más fácil realizar negocios son aquellas cuyos gobiernos cuentan con un sistema regulatorio "inteligente", con reglas diseñadas para ser eficientes, accesibles a todo el que necesite recurrir a ellas, sobre todo las que son sencillas de aplicar y de bajo costo.

Mientras que en México las empresas dedican 347 horas al año en promedio para pagar impuestos, en países como Noruega sólo dedican 87 horas. Además, México es uno de los países peor evaluados en estado de derecho lo que afecta los tiempos para ejecutar contratos por la vía legal (400 días con un costo de 31% del reclamo) mientras que en Singapur toma la mitad del tiempo.

Lo anterior desafortunadamente crea enormes oportunidades para la corrupción, considerada el principal obstáculo para hacer negocios en el país por 48% de los empresarios entrevistados por el Banco Mundial en el reporte Doing Business.

Recomendaciones:

- Establecer una ventanilla única digital para crear una empresa (siguiente versión de tuempresa.gob.mx).
- Simplificar trámites como el pago de impuestos con tutoriales en línea y declaraciones pre-llenadas.
- Generar tribunales especializados en temas de asuntos de propiedad intelectual y comercio electrónico.

Nota del editor: La Cámara de Diputados aprobó el 9 de febrero la figura denominada Sociedades por Acciones Simplificadas (SAS). Entre los beneficios que promete está el registro de empresas en un solo día, a costo cero y por internet. Esperamos que esto dé buenos resultados y mejore significativamente el entorno para hacer negocios.

INFRAESTRUCTURA DE APOYO

Aunque hace 10 años se impulsó la creación de incubadoras en México resultando en cerca de 500 instituciones en todo el país al 2012, sus resultados no han sido los esperados. Los expertos coinciden que el modelo de incubadoras ha fallado por dos principales causas: (1) las incubadoras no agregan valor, pues sus directivos y consultores carecen de experiencia empresarial y (2) no cuentan con fuertes vínculos con el sector productivo. La política de financiamiento de las incubadoras es perversa pues depende del número de empresas que creen, independientemente de su éxito. Solo una de cada siete incubadoras da algún tipo de algún sistema de seguimiento de sus empresas. Lo anterior explica por qué 70% de los emprendedores que han conseguido financiamiento en el mercado no provienen de alguna incubadora.

Recomendaciones:

- Cambiar los incentivos a las incubadoras a través de otorgar recursos públicos con base en el impacto que tengan sus empresas (empleos, ingresos, etc.). En este sentido, el primer paso es transparentar y evaluar los fondos erogados por la incubadora y limitar los apoyos a periodos cortos (como de un año) para evitar la dependencia de las empresas con la incubadora.

- Crear programas de mentoría en etapas tempranas de las empresas.
- Realizar auditorías profesionales desde el primer año de existencia para establecer la gobernabilidad y salud financiera de las empresas.

CAPITAL HUMANO

De acuerdo a encuestas realizadas por Select al sector TIC, 66% de las Pymes TIC tiene dificultad para contratar personal calificado (56% para todas las empresas). Uno de los principales problemas es el bajo nivel de dominio del idioma inglés. A causa del modelo unidimensional de educación a nivel licenciatura en México, los ingenieros en sistemas aunque cuenten con una solidez técnica carecen de elementos de negocios como marketing, funcionamiento de mercados y acceso a capital. Igualmente, los estudiantes de negocio carecen de las habilidades técnicas para llevar a cabo un negocio en TIC. Lo anterior se agrava porque México no facilita ni busca atraer talento TIC del mundo; al contrario, se ve como una competencia poco deseable para los mexicanos.

Existen múltiples pendientes en materia educativa en el país que contribuirán a mejorar este factor pero entre las principales acciones que IMCO propone están:

- Alinear programas educativos con necesidades de emprendedores.
- Flexibilizar carreras de ingeniería para promover más habilidades de negocios.
- Facilitar la entrada de talento a México como visas para emprendedores e investigadores.

FINANCIAMIENTO

Pese a los avances del sector (53% de las empresas mexicanas entrevistadas considera que el acceso a financiamiento ha mejorado en los últimos cinco años) los emprendedores entrevistados lo consideran el segundo factor más rezagado. Por un lado, el mercado de capital de riesgo es mucho menor a su potencial. En México se invierte 0.06% del PIB en dicho mercado mientras que en Colombia y Chile se invierte el doble en términos relativos. Por otro lado, el mercado de crédito a las empresas es limitado. En 2010, el porcentaje de crédito a empresas representó 21% del PIB mientras que el promedio latinoamericano fue 36%. Como resultado, los emprendedores tienden a financiar sus proyectos con fondos personales y familiares (58% de los emprendedores en TIC declaran haber utilizado fondos personales). Las razones de estos resultados se encuentran en: las limitaciones de los emprendedores a entender los requerimientos del sector financiero de contar con estados financieros completos, gobernanza corporativa y transparencia financiera. También se explica por la poca participación de individuos con alta capacidad de inversión en el mercado de capital emprendedor. Los empresarios mexicanos prefieren invertir en sectores tradicionales que en nuevos mercados, lo cual ha inhibido la creación de inversionistas ángel. Otra barrera son las limitadas salidas a mercado, por ejemplo, los requisitos para realizar una oferta pública son complicados y están dirigidos a empresas de mayor tamaño.

Recomendaciones:

- Introducir esquemas de financiamiento diseñados para cada etapa de desarrollo de las empresas. A la par se puede trabajar con la Bolsa Mexicana de Valores (BMV) para generar esquemas de ofertas públicas para un mercado de menor escala.
- También se puede utilizar un scoring psicométrico como mecanismo para el otorgamiento de créditos al menos en los fondos públicos y así contrarrestar la falta de colateral e historial crediticio.
- Asegurar que las inversiones de los fondos vengán acompañadas de capacitación y mentoría de calidad.

PERFIL DE LA DEMANDA

Este factor se refiere a la capacidad de un país para adoptar y promover el consumo de emprendimientos TIC. Su relevancia se debe a que aquellas pequeñas empresas que más crecen son las que usan de forma permanente las TICs para relacionarse con sus clientes de acuerdo a un estudio reciente de BCG. Sin embargo, el comercio electrónico en Brasil representa tres veces el de México en relación al PIB, debido principalmente a: el alto costo y baja calidad del acceso a internet, la baja penetración de servicios bancarios, y la poca confianza en el comercio electrónico.

Recomendaciones:

- Aumentar los acuerdos bilaterales para que las empresas mexicanas cumplan con los requisitos de comercio electrónico de otros países.
- Adoptar estándares internacionales de seguridad y privacidad de datos personales alrededor de internet.
- Abrir el Servicio Postal Mexicano a la competencia.

INTEGRACIÓN DE LA INDUSTRIA TIC.

Aunque en términos generales la industria TIC se encuentra integrada bajo 3 principales asociaciones y 38 clústers en 28 estados, esto no necesariamente se refleja en las nuevas empresas. De acuerdo con la encuesta realizada por IMCO, solo 23% de los emprendedores TIC pertenece a una asociación empresarial y 28% a un clúster. Pese a los logros de Prosoft y la mayor integración que perciben los empresarios de los clusters (ITAM) no hay mecanismos para vincular empresas que se están formando con la comunidad empresarial TIC. Los mecanismos privados de asociación son caros para los emprendedores por las cuotas y aunque lo pudieran pagar, no encuentran sus intereses representados. Para los emprendedores entrevistados no está claro que se tengan métricas de éxito de los clústers de software y más bien encuentran a la mayoría de éstos desactualizados.

Recomendaciones:

- Hacer planes regionales para los clústeres de nueva generación y nichos de especialización que aseguren que éstos cuentan con los requisitos básicos para su éxito.
- Facilitar que los emprendedores se inserten en la cadena de valor de un clúster por medio de incentivos fiscales.

CULTURA EMPRENDEDORA

De acuerdo a un estudio académico (Ardagna & Lusardi 2008), este factor es uno de los que mejor explica la probabilidad de emprender en un país. Si los individuos tienen confianza en sus

habilidades empresariales y conocimientos, tienen 8.5% de mayor probabilidad de emprender un negocio.

Además, aquellos que no le tengan miedo al fracaso tienen una probabilidad 10% mayor a emprender. Finalmente, el conocer a un emprendedor aumenta la probabilidad de emprender en 3%. En México esta cultura no se ha consolidado. En una encuesta realizada por Grupo Monitor (Deloitte) el porcentaje de individuos que considera empezar un negocio como una buena alternativa es 58%, mientras que en Brasil es de 80%. Las razones, según los expertos, es que no existe la cultura de mentoría en México entre los empresarios más experimentados debido a la falta de interés y falta de mecanismos y eventos para que interactúen. Por otro lado, no hay visibilidad de los casos de éxito de emprendedores en la sociedad. Otra característica de México es que no existe una cultura de riesgo ni de fracaso y no ha quedado claro qué es ser emprendedor. De acuerdo a encuestas de la Secretaría de Economía sobre incubadoras, no hay claridad sobre los tipos de sociedades mercantil óptimas, rondas de negociación con un inversionista, o maneras de presentar un proyecto.

Recomendaciones:

- Aumentar la promoción de sitios web gubernamentales y redes sociales para los emprendedores exitosos.
- Facilitar eventos o concursos como los Hackatones o pasarelas de emprendedores.
- Otorgar becas para emprendedores de menores ingresos para continuar con sus proyectos.

ESTRATEGIA GUBERNAMENTAL

Este es el factor que se encuentra en mejor estado de acuerdo a los expertos entrevistados por IMCO. En particular se reconoce la labor de la política pública en crear apoyos a fondos logrando que se duplicaran en 6 años, crear los primeros fondos de capital semilla en el país y vincular a las universidades públicas con proyectos emprendedores a través de encuentros entre emprendedores y el sector. Muestra de esto es que 73% de los emprendedores encuestados declara haber asistido a un evento organizado por el gobierno. Sin embargo, aún quedan importantes retos que resolver: 93% de los emprendedores encuestados considera que hay demasiados programas de apoyo y son complicados de acceder; prueba de ello es que hay gestores que pueden cobrar por asesorar para conseguir recursos. Por otro lado, se encontraron 133 programas para investigación y desarrollo pero 97% de los emprendedores considera que no se ajustan a sus necesidades. Además 88% de los emprendedores señala que la información de los programas no se entiende fácilmente y que los recursos llegan de forma lenta. Otro aspecto que señalan es la falta de seguimiento a programas y mecanismos de salida lo que puede generar dependencia a dichos programas.

Recomendaciones:

- Fortalecer y mejorar el sitio web creado por el INADEM en el cual se concentran algunos de los recursos públicos y privados para emprendedores.
- Diferenciar las políticas por etapa de emprendedor y evaluar dichos programas a través de un tercero cuyos resultados serían públicos. ☺

Expertos en desarrollo de soluciones TI a la medida, bajo metodologías y estándares de calidad.



Desarrollo de **Software a la Medida**

Personal de TI bajo **Outsourcing+**

Desarrollo de **Apps Móviles**



UNIFACE.COM
Partner

★ Training y Certificaciones



- *Android básico
- *Android experto
- *Android wear
- *iOS-SWIFT básico



- *Redes y Servidores
- *Proyectos de SW

Consulta fechas en cursos.dwsoftware.mx

Industria de Software en **Colombia**

Por Alejandra Jiménez



Continuando nuestra serie de reportajes sobre la industria de software en distintos países de Latinoamérica, hemos decidido dedicar este número a estudiar el caso de Colombia.

Todos los que nos dedicamos al software hemos visto la importancia que ha estado cobrando poco a poco este país en nuestra industria. Ya sea como un mercado de servicios de tecnología, un lugar desde donde hacer outsourcing aprovechando el talento local, o una fuente de startups tecnológicas, Colombia se ha estado ganando la mirada de muchos de nosotros a través de los últimos años.

Conozcamos entonces un poco sobre cómo es que la industria de TIC de Colombia ha llegado hasta aquí, y qué es lo que sigue.

MINTIC: IMPULSANDO LA INDUSTRIA DESDE EL MÁS ALTO NIVEL

Colombia tiene el caso particular y notable de contar con un Ministerio (lo

que equivaldría a una Secretaría en México) dedicado únicamente al sector de Tecnologías de Información y Comunicaciones [1].

El Ministerio de Tecnologías de la Información y las Comunicaciones (MinTIC), es la entidad que se encarga de diseñar, adoptar y promover las políticas, planes, programas y proyectos de este sector.

El Plan Vive Digital [2] fue lanzado en 2010 para impulsar la masificación del uso de Internet con el fin de reducir la pobreza y generar empleo. Ha representado un gran salto tecnológico a nivel nacional, involucrando a todas las entidades gubernamentales del país. Los 4 ejes de este plan, y que actualmente se siguen promoviendo, son: aplicaciones, usuarios, servicios e infraestructura.

Actualmente el plan se encuentra en su segunda iteración (2014-2018) y entre sus metas, las siguientes son algunas de las que más nos llaman la atención:

- Generar 369 mil empleos en TI.
- Ubicar al menos una ciudad de Colombia en el Top 25 del mundo para negocios TIC.
- Certificar como ciudadanos digitales a 2,5 millones de funcionarios públicos para que se apropien de la tecnología.
- Impulsar teletrabajo. La meta es llegar a los 120 mil teletrabajadores.
- Enfocar el servicio social obligatorio para que los estudiantes enseñen a sus comunidades sobre las nuevas tecnologías.

Apps.co [3] es la iniciativa de MinTIC enfocada en apoyar el emprendimiento tecnológico. Para lograr esto ofrece servicios y programas para atender algunas de las principales necesidades de los emprendedores, tales como: ayudarlos a desarrollar sus ideas de emprendimiento, conectar con inversionistas, ayudar a difundir sus productos/servicios, y brindar acceso a recursos de capacitación en línea.

Alejandra Jiménez es editora y escritora enfocada en temas de tecnología y emprendimiento. Facilitadora de Startup Weekend desde 2012. Ha sido colaboradora de publicaciones como PulsoSocial y Fayerwayer.

Colombia no es ajeno a la necesidad global de más y mejor talento en desarrollo de software, y para ello MinTIC cuenta con la iniciativa Talento Digital [4], con la que se busca formar a miles de profesionales en TI. Por medio de este programa se ofrecen créditos condonables en distintos porcentajes en el pago de colegiaturas, buscando con ello atraer a más jóvenes a las carreras relacionadas con TI. De hecho, a esta iniciativa se han destinado recientemente 96.3 millones de dólares, que serán distribuidos en un período de 3 años (2015-2018) para beneficiar a los ciudadanos colombianos interesados en formación académica en carreras relacionadas con TI.

COLOMBIA DIGITAL

Es una corporación que surge en 2002 para promover el uso y apropiación de las tecnologías de información y las comunicaciones, en beneficio del desarrollo social y económico. Su trabajo se basa en 3 enfoques:

- Fortalecer la adopción y uso de TIC en la sociedad, orientados al mejoramiento de su calidad de vida.
- Promover la competitividad de las organizaciones públicas y privadas, utilizando las TIC.
- Apoyar la formulación, implementación y apropiación de políticas TIC, en escenarios de gobierno colombiano y global.

Son una organización de carácter mixto cuyos fundadores provienen de los sectores académico, empresarial y gobierno. Hoy, Colombia Digital es una corporación independiente que no depende de ninguna instancia pública o privada, y actúa bajo un modelo de auto-sostenibilidad basado en la gestión de proyectos

FEDESOFTE

La Federación Colombiana de la Industria del Software y Tecnologías Informáticas Relacionadas (Fedesoftware) es el gremio de la industria de TI en el país, con 27 años de existencia y 410 empresas afiliadas hasta el momento, en su mayoría nacionales. Ximena Duque, Directora de Empresas Competitivas dentro de la Federación, nos comenta que son 4 las áreas de trabajo:

1. Empresas competitivas. Enfocado en medir índices de competitividad, ventas, que sus empresas cuenten con

estándares internacionales, ayudar a identificar necesidades de la industria en formación, planeación para convertirlas en metas reales.

2. Posicionamiento global. Dedicado a posicionar a la industria colombiana como productor de software, facilitar a sus empresarios participación en ferias, entrenamientos gerenciales —no solo dentro de Colombia sino en otros países.

3. Talento humano. Divulgar actividades de formación técnica y profesional, procesos de calidad y asegurar que esta información llegue al sector.

4. CENIsoft. Centro de Investigación especializado en la industria de software, creado por Fedesoftware para que sus afiliados desarrollen estudios e investigaciones, establezcan alianzas con otros organismos e instituciones educativas y se genere un observatorio de TI.

Ximena comenta que: “la industria de TI está creciendo, porque la infraestructura y el gobierno lo han facilitado, y a su vez esto genera una mayor demanda de productos y servicios de TI.”

Gobierno y FEDESOFTE apuestan por TI bajo la premisa de que esta actividad es transversal en la economía, y es por esto que en los programas de la Federación se busca una articulación con iniciativas lanzadas desde MINTIC o el Ministerio de Educación.

EL CASO DE MEDELLÍN

Aunque Bogotá es la capital y ciudad más grande de Colombia, reuniendo cerca de 10 millones de habitantes en su zona metropolitana, es en Medellín donde hay más dinamismo en términos de innovación y emprendimiento tecnológico. A nivel mundial es considerada como una de las ciudades con los mejores programas para apoyar la generación de nuevas empresas y startups, y en 2013 obtuvo el reconocimiento por parte del Wall Street Journal como la ciudad más innovadora del mundo, por encima de los finalistas Nueva York y Tel Aviv. Medellín es, sin duda, un referente cuando se habla de tecnología e innovación.

Destaca el caso de Ruta N, una corporación creada por la Alcaldía de Medellín con conjunto con UNE y EPM (empresas locales proveedoras de servicios públicos y de telecomunicaciones), con el fin de

facilitar la evolución económica de la ciudad hacia negocios intensivos en ciencia, tecnología e innovación, de forma incluyente y sostenible. Su objetivo al año 2021 es posicionar a Medellín como la ciudad más innovadora de América Latina.

Otro buen punto de entrada al ecosistema local es el programa Medellín Emprendedora. Lo conforman un conjunto de entidades que apoyan a los emprendedores para realizar sus proyectos, impulsar y/o fortalecer emprendimientos. Es una de las apuestas del gobierno de Medellín por convertir a su ciudad en uno de los referentes de tecnología y creación de startups a nivel nacional y regional con proyección internacional.

En cuestión de espacios, destaca el Parque del Emprendimiento, un espacio físico de 1,000m² que ofrece servicios de hospedaje y aceleración a nuevos emprendimientos de base tecnológica.

SALARIOS EN COLOMBIA

De acuerdo a una encuesta realizada recientemente por la comunidad Colombia Dev, el salario medio para un desarrollador de software en Colombia es de alrededor de \$1,100 dólares al mes. Al igual que en el caso de México, esto puede subir drásticamente dependiendo de la experiencia, el dominio del lenguaje inglés y las tecnologías utilizadas. Para mayor información recomendamos el análisis hecho por nuestros colegas de Make it Real [7].

CONCLUSIÓN

Colombia está decidida a posicionarse como jugadora clave a nivel mundial en la industria de software. Cuenta con una oferta de valor interesante, y parece estar avanzando a buen ritmo. Sin duda es una gran adición al equipo de “Latam IT”.

Referencias

- [1] <http://www.mintic.gov.co>
 - [2] <http://vivedigital.gov.co>
 - [3] <http://apps.co>
 - [4] <http://talentodigital.gov.co>
 - [5] <http://www.colombiadigital.net>
 - [6] <http://www.medellinemprendedora.net>
 - [7] <http://rutanmedellin.org>
 - [8] <http://parquedelemprendimiento.com>
 - [9] <http://blog.makeitreal.camp/>
- [analisis-encuesta-desarrolladores-colombia-2016](#)



● **Como parte de nuestro reportaje sobre Colombia**, compartimos aquí la opinión y perspectiva de personas que conocen este ecosistema de primera mano.

JUAN PABLO BURITICÁ

Juan Pablo es un desarrollador de origen colombiano. Actualmente es VP de ingeniería en ride.com basado en Nueva York y cuenta con un equipo de ingenieros trabajando desde Colombia. Adicionalmente es co-organizador de JSConf Colombia, cofundador de la comunidad Colombia Dev y frecuentemente escribe y habla sobre el desarrollo de talento y servicios digitales desde Colombia.

SOBRE EL (SUPUESTO) DÉFICIT DE INGENIEROS DE SOFTWARE

El problema nace en la industria, hay una brecha entre lo que la industria demanda y lo que buscan las nuevas generaciones. Desgraciadamente es común ver al ingeniero de software como algo cercano a un obrero, porque es visto desde la perspectiva de la maquila, sin innovación. Ese es el reto que tiene gobierno, satisfacer la demanda de empresas con tecnología, visión y sistemas obsoletos cuesta cada vez más trabajo. Si tú vas a una universidad en Colombia, digamos, al 1er semestre de la carrera de ingeniería en software o sistemas, nadie quiere trabajar en la industria local; el estigma contra la ingeniería es un problema de mercado, no técnico. Los ingenieros graduados quieren trabajar en empresas internacionales, muchos dirán Google, Facebook, pero otros tantos solo buscan lugares donde pueden sentir orgullo de crear, lugares donde exista una verdadera cultura de ingeniería, donde se pueda innovar. Un ingeniero busca ser ingeniero en su trabajo; las startups son eso, laboratorios de ingeniería con recursos limitados pero suficientes para trabajar en un problema específico y puntual por un espacio de tiempo.

Afortunadamente hay organizaciones en Colombia que están entendiendo lo que representa ofrecer una cultura de ingeniería y un ambiente en donde el talento local desearía trabajar, algunos ejemplos son: Koombea, S4N y Monoku, las tres encargadas de trabajar en soluciones digitales que van desde desarrollo, diseño hasta consultoría. Son ejemplos de que las empresas de consultoría también pueden hacerse interesantes aunque sirvan a un mercado de servicios.

SOBRE LOS ESFUERZOS DEL GOBIERNO PARA CONSTRUIR SERVICIOS DIGITALES

Creo que es importante aclarar que una cosa es TI y otra son servicios digitales. El lenguaje es erróneo, yo no hago TI, yo construyo servicios digitales.

El gobierno hace intentos, como por ejemplo, la implementación del Código Postal o servicios digitales, pero son demasiado difíciles de usar. Creo que no tiene personal requerido para construir este tipo de servicios de forma adecuada. Se requiere una mentalidad más abierta y transparente. Por ejemplo, en Estados Unidos hay una iniciativa de la Casa Blanca llamada U.S. Digital Services, que reconoce que la forma de interactuar con el gobierno es muy complicada y convoca a los ciudadanos para que se integren a equipos y apliquen técnicas de diseño de productos para definir y construir servicios digitales sencillos y útiles. Así que en ese sentido, siento que lo que le hace falta, más que un Ministro de TIC, es un CTO.

DANIEL ARISTIZABAL

Daniel es un artista gráfico y desarrollador colombiano. Junto a Juan Pablo Buriticá inició la primer comunidad de JavaScript en Colombia, BogotaJS, hace ya 4 años. Este formato se ha replicado ya en otras

ciudades del país (Medellín, Barranquilla, Pereira), en donde las reuniones mensuales permiten un intercambio de conocimientos entre sus participantes, y se abre espacio para que los miembros de la comunidad compartan sus experiencias. El impacto de estas comunidades se ha transformado en conferencias a nivel nacional. JSConf Colombia, con dos ediciones, ya es un evento de relevancia internacional.

SOBRE LA VISIÓN DE LAS NUEVAS GENERACIONES

La tecnología que tenemos en muchos de los servicios que se ofrecen a la ciudadanía están basados en softwares de gestión de hace 10-15 años. Dos o tres compañías en Colombia son las que se siguen contratando para soportar tecnología vieja con contratos millonarios; hay miedo ante las nuevas startups, con mejor tecnología y mejores visiones, que pueden hacer las cosas mejor y con costos menores. Las empresas tecnológicas nuevas y las startups, ya no ven coherencia en relacionarse, por ejemplo, con FEDESOFTE. Ahora el impulso a desarrollar software, con nuevas tecnologías se da, principalmente por parte de comunidades tecnológicas y estudiantiles, y las startups surgen porque son empresas enfocadas en resolver problemas muy puntuales. La tecnología ya no es solo el interés de unos pocos, o de los ingenieros de en sistemas, la tecnología ha encontrado espacios en todas las ramas. Veo en nuestras reuniones a artistas, abogados, activistas de derechos humanos, ya es un ecosistema diverso.

SOBRE LA EDUCACIÓN

La educación tradicional está basada en castigo e incentivo, se incentiva la inteligencia lógica, la memorización, y si no memorizas, castigo. No se desarrollan otras habilidades, no se enseña a manejar riesgos, ni hay un enfoque a otro tipo de inteligencias, como

la emocional, la financiera o la social. Aun así, creo que algunas iniciativas promovidas desde gobierno como Apps.co y ViveLab son interesantes para quienes desean emprender y acercarse a la tecnología.

ALEXANDER RICAND

Alexander es Director del Departamento Administrativo TIC de la Alcaldía de Tuluá. Considera que el ingenio y talento colombiano se ha destacado en varias áreas del conocimientos, siendo la industria del software una de ellas. A pesar de ello, considera que la escasez de talento suficiente es un problema real y opina: "se estima que el país para 2018 tendrá un déficit de Talento TI superior a los 80,000 ingenieros y profesionales en la industria, hecho que preocupa dadas las condiciones competitivas en la región".

SOBRE EL ROL DEL GOBIERNO

El Gobierno en todos sus niveles, puede incidir en la generación de nuevos y mejores escenarios para la industria TI. Por supuesto, hay que tener claro que su papel debe ser de articulador, en ningún momento puede ser

considerado como el actor principal, dado que si la política pública sufre cambios o considera que los objetivos deben ser otros; todo el sector podría verse afectado. Los protagonistas deben ser las personas, las comunidades y las organizaciones, teniendo como propósito el beneficio común.

Considero que se debe continuar trabajando en dos importantes elementos. El primero es la generación de más escenarios para que en etapas tempranas quienes tienen habilidades, talentos e intereses en la industria TI se sigan conectando y explorando sus habilidades, por eso eventos como Hackatones, Startup Weekend, entre otros son importantes.

El otro elemento es la consolidación de talentos en etapa madura para el diseño, servicio y soporte de soluciones para el mercado global, pues en la actualidad son pocas las empresas locales que cuentan con infraestructura y experiencias en prestación masiva de servicio y cuando hablamos de masivo nos referimos a grandes porciones de usuarios finales.

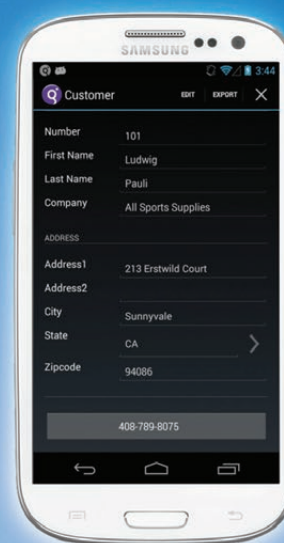
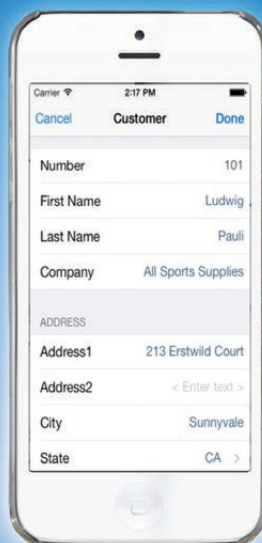
EDWARD RAMÍREZ

Edward Ramírez es fundador de Partners 4 Startups (p4s.co), una comunidad que conecta a personas de distintas especialidades interesadas en el emprendimiento tecnológico. Con el objetivo de dar visibilidad a la comunidad tecnológica, en 2015 compitió por una posición en el Concejo de Bogotá.

SOBRE EL ENFOQUE DEL GOBIERNO

Es crucial explorar cómo se pueden coordinar los esfuerzos entre el gobierno y las comunidades independientes. También es importante cambiar un poco la orientación de objetivos políticos, debido a que actualmente se termina apoyando a apuestas de corto plazo, inmediatas, dejando de lado apuestas más ambiciosas que requieren mayor trabajo y tiempo. Adicionalmente, el Gobierno está incentivando al sector como nunca antes; existen muchos programas de apoyo pero ante la comunidad no se logran visualizar de forma integrada, parecieran esfuerzos dispares que persiguen objetivos independientes.

Un framework. Un código. Un ciclo de prueba. Menos trabajo y mas playa!



Reduzca el tiempo de liberación creando aplicaciones nativas elegantes, para iOS y Android a la vez. Ya sea para trabajar de manera autónoma, o en la nube, todo en un solo ciclo de desarrollo.



Lo esperamos en
SG Mobile Day 2016
Mar 16. Crowne Plaza WTC



Four Js Development Tools Latinoamérica, SA de CV
Insurgentes Sur 1602 México DF 03940
Tel +52 (55) 1000 9160
www.4js.com - latin.4js.com

1 WEBVR



WebVR es un API de Javascript que permite que aplicaciones web puedan interactuar con dispositivos de realidad virtual, como el Oculus Rift o Google Cardboard. De esta manera, una aplicación web podría recibir eventos o información generados por el visor, y de la misma manera enviar eventos e información que alimenten escenarios 3D en el visor. Esto se puede aplicar a diversos escenarios tales como tours virtuales, entrenamiento interactivo, y juegos altamente inmersivos.

WebVR es un proyecto open source respaldado por Mozilla. Recientemente liberó su versión 1.0, y es soportado en el nightly build más reciente de Firefox y Chrome.

Más información en <http://mozvr.com>

2 MANTL



Mantl es una plataforma para gestionar la instalación y operación de microservicios en infraestructura distribuida. El enfoque de Mantl es brindar una solución completa, que resuelva todos los aspectos de gestión de microservicios, de manera que los desarrolladores no tengan que estarse preocupando por la infraestructura y pueda enfocarse en la funcionalidad de los servicios que construyen. Mantl está basado en estándares y opera con distintos proveedores y tecnologías tales como Openstack, AWS, Vagrant, o directamente en el hardware (bare metal).

Mantl es open source y es desarrollado y soportado por Cisco Systems, como parte de sus esfuerzos para adentrarse en el ecosistema DevOps.

Más información en <http://mantl.io>

3 KOTLIN



Jetbrains, la empresa creadora de IDEs como IntelliJ y TeamCity, liberó la versión 1.0 de Kotlin, un lenguaje de programación para la máquina virtual de Java.

Kotlin es descrito como un lenguaje "pragmático", para aquellos desarrolladores que buscan un lenguaje con capacidades modernas para mayor productividad, pero que sea compatible con el resto de la plataforma Java. Kotlin combina capacidades de orientación a objetos y programación funcional, y está enfocado en la interoperabilidad, estabilidad e integración con herramientas.

Kotlin es open source y se ha estado desarrollando desde hace más de 5 años.

Gracias a su completa compatibilidad con Java, Kotlin parece ser una buena alternativa para desarrollar aplicaciones Android.

Más información en <https://kotlinlang.org>

4 SQL SERVER 2016



Se requiere bastante para que en pleno 2016 nos podamos entusiasmar por un servidor de base de datos relacionales, pero debemos confesar que SQL Server 2016 lo está logrando. Entre las capacidades que incluirá destacan:

- Podrás utilizar R directamente en la base de datos para realizar análisis estadístico, sin necesidad de estar exportando los datos.
- Stretch Database, permite que una misma base de datos tenga algunas particiones almacenadas localmente y otras en la nube (via Azure SQL Database) La intención sería tener los datos de mayor uso localmente y los datos que son accedidos con menos frecuencia en la nube, para mitigar costos de almacenamiento.
- Polybase, el conector para Hadoop incluido en Analytics Platform System, también está en SQL Server 2016.
- Soporte a JSON, facilitando el almacenamiento y manipulación de datos con este formato.

SQL Server 2016 actualmente se encuentra disponible como Community Technology Preview y se espera que la versión para producción sea liberada antes de la mitad del año.

El Fin de la Era del “Big Data”

Por Luis Daniel Soto Maldonado

● **Las decisiones** se están evaporando de nuestra vida: hoy nos encontramos utilizando apps y el internet para decidir qué libro leer, en dónde cenar, qué película ver en el cine, qué programa ver en televisión, qué música escuchar, en qué acciones invertir y a qué amigos felicitar. Las máquinas de búsqueda deciden cómo comenzaremos a navegar la red. Una tercera parte de los matrimonios en Estados Unidos se originan por citas en línea, un algoritmo inteligente y aprendizaje de máquina decide quiénes son “compatibles”. Las máquinas ya hacen más de lo que creemos que son capaces.

El cáncer no se remediará con un medicamento milagroso, no es un solo tipo. El problema de esta terrible enfermedad posiblemente se resolverá con aprendizaje de máquina. Se requiere un sistema inteligente, capaz de entender cuál es el medicamento apropiado para cada persona, capaz de conocer profundamente los hábitos del enfermo, sus antecedentes familiares, su DNA, su historia médica, las mutaciones de la célula, los resultados de otros tratamientos.

Es así que la era de “big data” finalmente está llegando a su fin. No es porque se haya implementado en la mayoría de las empresas, o porque las mismas ya hayan descubierto la riqueza de tomar decisiones verdaderamente informadas y en tiempo casi real. No, lo que sucede es que estamos entrando a una nueva etapa en donde el uso de los datos permite el verdadero aprendizaje de máquina. El conocimiento se empieza a acumular: aunque estamos distantes del auto que se conduce por sí mismo en un ambiente real, los vehículos autónomos que se están desarrollando cuentan ya millones de horas de aprendizaje, realizando la tarea de aprender haciendo.

Reflexionemos cómo hemos llegado aquí. La vida mecanizada no es lo que se pensó algunas décadas atrás. Los empleos del conocimiento están siendo más fáciles de automatizar que los de labor manual, como el trabajador de construcción. Todos esperamos robots que puedan verdaderamente realizar labores domésticas, pero esto conlleva implicaciones considerables como la pérdida de capacidades básicas del hombre, gracias a estos mismos avances.

Esta situación se presenta hoy en la industria médica y la aeronáutica, con una sobre-dependencia en sistemas informáticos. Si no hay sistema, no hay capacidad de actuar. Nunca habíamos alcanzado los actuales niveles de seguridad y confiabilidad en seguridad aérea, pero los pilotos tienen control manual de un avión solo unos pocos minutos por vuelo. Aunque hay beneficios, los pilotos están perdiendo sus habilidades básicas por el exceso de automatización. En situaciones de emergencia, no son capaces de recuperar el control y fallan en evitar accidentes.

Un debate que inició hace dos siglos está recientemente cobrando mayor relevancia: la separación entre hombre y máquina. ¿Cuándo una máquina será capaz de escribir una obra de arte? Las computadoras de hoy pueden, con facilidad, juzgar el éxito de un libro. Pero, ¿por qué no escribir un “Don Quijote”, “Las Mil y una Noches” o “Cien Años de Soledad”? Estamos en ese camino. Creo que las computadoras ofrecerán cientos de puntos de partida para que la creatividad humana llegue mucho más allá.

Los algoritmos y robots súper-inteligentes nos expondrán a sentimientos que nunca anticipamos por las máquinas. Hoy son solo búsquedas de texto muy complejas, pero la búsqueda por el algoritmo del aprendizaje de cualquier cosa es el problema clave a resolver. Esos sistemas llegarán con nuevas implicaciones éticas, políticas y sociales.

Regresemos a nuestro punto de partida. Empecemos por definir cómo evaluaremos dichas tecnologías, quien estará a cargo de la nueva Inteligencia Artificial en la organización y qué implicaciones brindan a nuestros modelos de negocio. Aún no hemos visto todos los actos de magia de la tecnología. ☺

Si te interesa leer más sobre máquinas y algoritmos inteligentes, te recomiendo que visites el compendio de artículos que he creado en Flipboard: <http://swgu.ru/re>



Luis Daniel Soto (@luis-dans / @luisdanielsoto) trabaja en Amazon Web Services, enfocado en el desarrollo global de negocios para Big Data e Inteligencia de negocios. sotols@amazon.com



Un Vistazo a Apache Spark Streaming

Por Pedro Galván

● **Apache Spark** es un framework open source para el procesamiento de datos masivos diseñado con tres prioridades en mente: velocidad, facilidad de uso, y capacidades avanzadas de analítica.

Spark está cobrando gran popularidad porque viene a resolver varias de las limitaciones inherentes de Hadoop y MapReduce. Spark puede utilizarse junto con Hadoop, pero no es requisito. Spark extiende el modelo MapReduce para hacerlo más rápido y habilitar más escenarios de análisis, como por ejemplo *queries* interactivos y procesamiento de flujos en tiempo real. Esto es posible ya que Spark usa un cluster de cómputo en memoria (*in-memory*).

Uno de los elementos clave de Spark es su capacidad para procesamiento continuo (*stream processing*). Esto se logra por medio del componente Spark Streaming. En este artículo brindamos un vistazo de cómo funciona Spark Streaming.

Descripción general

Spark Streaming puede ingerir datos de un amplio de fuentes, incluyendo flujos provenientes de Apache Kafka, Apache Flume, Amazon Kinesis y Twitter, así como de sensores y dispositivos conectados por medio de sockets TCP. También se pueden procesar datos almacenados en sistemas de archivos como HDFS o Amazon S3.

Spark Streaming puede procesar datos utilizando una variedad de algoritmos y funciones tales como map, reduce, join y window. Una vez procesados, los datos son enviados a archivos en file systems o para poblar dashboards en tiempo real.

A grandes rasgos, lo que hace Spark Streaming es tomar un flujo de datos continuo y convertirlo en un flujo discreto —llamado DStream— formado por paquetes de datos. Internamente, lo que sucede es que Spark Streaming almacena y procesa estos datos como una secuencia de RDDs (Resilient Distributed Data). Un RDD es una colección de datos particionada (distribuida) e inmutable. Es la unidad de información que el motor de procesamiento de Spark (Spark Core) tradicionalmente consume. Así

que cuando usamos Spark Streaming para alimentar un stream a Spark Core, éste último los analiza de forma normal, sin enterarse de que está procesando un flujo de datos, porque el trabajo de crear y coordinar los RDDs lo realiza Spark Streaming.

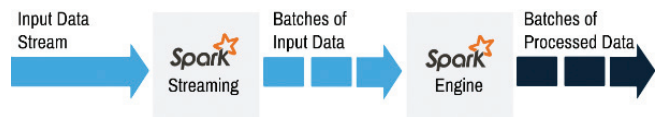


Figura 1. Spark Streaming prepara un flujo para que Spark Core lo pueda consumir.



Figura 1. Spark Streaming genera RDDs por intervalos de tiempo.

Orquestación del cluster

La figura 3 muestra a grandes rasgos cómo opera la orquestación del cluster. Las actividades son orquestadas por un programa conocido como Driver. Este programa típicamente instancia a SparkContext para realizar la orquestación de los procesos ejecutores, que son los que operan sobre los datos.

En el caso de aplicaciones basadas en streams, se utiliza StreamingContext en lugar de SparkContext, ya que StreamingContext soporta DStreams.

Como podemos ver, el Driver es un punto central de operación, así que es importante mitigar las posibilidades de que falle. Spark Streaming soporta un concepto llamado "checkpointing" que asegura que todos los datos y metadatos asociados con RDDs que forman flujos de datos son replicados continuamente a un almacenamiento tolerante a fallas. Esto permite que en caso de una falla del driver se puedan recuperar y procesar el flujo de datos.

Pedro Galván Kondo es cofundador y director editorial de Software Guru.

Estrictamente, Spark Streaming no opera en base a flujos continuos, sino a micro-batches.

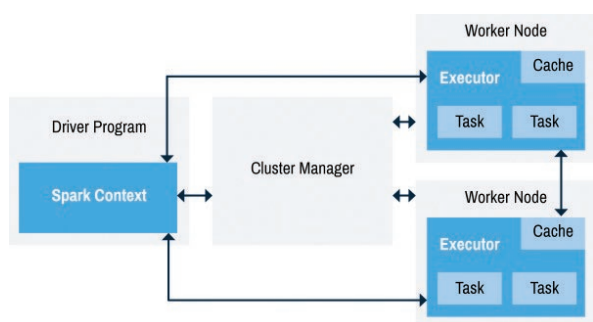


Figura 3. El driver orquesta el cluster de procesamiento.

Modelos de procesamiento

Spark Streaming soporta distintos modelos correspondientes a las semánticas típicamente utilizadas para el procesamiento de flujos. Esto asegura que el sistema entrega resultados confiables, aún en caso de fallas en nodos. Los flujos de datos pueden ser procesados de acuerdo a los siguientes modelos:

- Exactamente una vez (*exactly once*). Cada elemento es procesado una sola vez.
- A lo más una vez (*at most once*). Cada elemento puede ser procesado máximo una vez, y es posible que no sea procesado.
- Por lo menos una vez (*at least once*): Cada elemento debe ser procesado por lo menos una vez. Esto aumenta la posibilidad de que no se pierdan datos pero también es posible que se generen duplicados.

No todos estos modelos son soportados con todos los tipos de fuentes de datos. Es necesario verificar cuáles semánticas se soportan en cada una.

Desde un punto de vista de procesamiento, el modelo más sencillo de construir es "at most once". Lo que implica este escenario es que es aceptable que ocasionalmente haya pérdida de datos, ya que lo que más importa es mantener la continuidad del flujo. Pensemos en cómo funciona un stream de video: de vez en cuando se pierden paquetes de información y baja un poco la calidad, pero lo importante es que se mantenga el flujo y que no tengamos que comenzar desde el inicio.

Bajo un modelo "at least once", tenemos la garantía de que aunque haya alguna falla en algún nodo, no perderemos datos ya que cuando el nodo se recupere (o se reasigne su carga a otro) éste procesará todos los datos para asegurar que no se le vaya ninguno. Trasladando esto a nuestro ejemplo de streaming de un video, lo que sucedería es que todos los datos con buena calidad pero corremos el riesgo de que se repitan pedazos que ya habíamos visto. Al usar este modelo debemos buscar que las operaciones en nuestro código sean idempotentes, es decir que siempre produzcan el mismo resultado sin importar si han sido ejecutadas anteriormente (por ejemplo, $x = 4$ es idempotente pero $x++$ no lo es). De esta manera, no importa si procesamos un dato varias veces, ya que en ambas ocasiones nos generará el mismo resultado y podemos filtrar en base a esto.

Aunque el modelo "exactly once" es lo que parecería lógico que siempre debamos escoger, en realidad no es necesario en todas las ocasiones y hay que tener cuidado al escogerlo. Debemos entender que este modelo es el más intensivo en recursos y puede ocasionarnos problemas de desempeño debido a todo el procesamiento adicional que se requiere para asegurar que cada uno de los datos no se pierda ni se duplique. Adicionalmente, si nuestro código es idempotente, con un modelo at least once tendremos resultados confiables.

Consecuencias de micro-batches

Como ya vimos, estrictamente Spark Streaming no opera en base a flujos continuos sino a micro-batches que tienen un tiempo de intervalo entre ellos (típicamente de menos de 5 segundos). Es importante que entendamos las consecuencias que esto puede tener. Por un lado, se puede configurar y reducir el intervalo a menos de un segundo, lo cual nos daría un desempeño casi de tiempo real, pero con un alto costo en recursos de procesamiento. Adicionalmente, un argumento en contra del esquema de micro-batches es que puede ser que los datos no se reciban en el orden exacto en el que sucedieron. Esto puede o no ser relevante dependiendo de la aplicación específica. Por ejemplo, en un timeline de Twitter tal vez no sea indispensable que los tweets sean procesados exactamente en el mismo orden en el que fueron generados. ☹

Referencias

[1] J. Scott. "A Quick Guide to Spark Streaming". <http://swgu.ru/rf>



¿QUÉ ES UNA ORGANIZACIÓN DIRIGIDA POR DATOS?

Por DJ Patil y Hilary Mason

Este artículo es un extracto del libro "Data Driven" escrito por DJ Patil y Hilary Mason y publicado por O'Reilly Media.

● **Las organizaciones** más conocidas que están dirigidas por datos, son empresas nativas de internet como Google, Amazon, Facebook y LinkedIn.

Sin embargo, estar dirigido por datos no está limitado a internet. Walmart ha sido pionero en el uso de datos desde los 70. Fue una de las primeras organizaciones en construir grandes data warehouses para gestionar el inventario a través de sus negocios. Esto la habilitó para convertirse en la primera empresa en tener más de 1 billón de ventas durante sus primeros 17 años. Y la innovación no terminó ahí. En los 80, Walmart se dio cuenta que la calidad de sus datos no era suficiente, así que para adquirir mejores datos se convirtió en la primera empresa en utilizar lectores de código de barra en las cajas. La empresa quería saber qué productos se vendían y cómo es que el acomodo de productos en la tienda impactaba las ventas. También

requería entender tendencias estacionales y diferencias regionales. Conforme aumentó el número de tiendas y volumen de productos, también aumentó la complejidad de manejar el inventario. Gracias a la disponibilidad de datos históricos combinados con un modelo predictivo, la empresa pudo controlar su curva de crecimiento. Para reducir aún más el tiempo para que sus datos se convirtieran en decisiones, Walmart fue la primera empresa en invertir en tecnología RFID. Y recientemente ha puesto sus esfuerzos en tecnologías de procesamiento de datos como Hadoop y Cassandra.

FedEx y UPS también son conocidos por utilizar datos para competir. Por medio de análisis de datos, UPS se dio cuenta que si sus choferes solo daban vueltas hacia la derecha (evitando las vueltas a la izquierda) tendrían menos accidentes, menos tiempo perdido, y ahorrarían en

gasolina. Los resultados fueron sorprendentes, en un año ahorraron alrededor de 10 millones de galones de gasolina, contribuyendo también así, a disminuir su impacto al medio ambiente.

De forma similar, General Electric utiliza datos para mejorar la eficiencia de sus motores de avión. Actualmente hay aproximadamente 20 mil aviones operando con 43 mil motores GE. Durante los próximos 15 años se espera que se utilicen 30 mil motores más. Una mejora en eficiencia de 1% resultaría en 30 billones de dólares a través de los próximos 15 años. Parte del esfuerzo para atacar estos problemas es el nuevo motor GEnx. Cada motor cuenta con más de 4 mil partes y pesa cerca de 6 toneladas. Uno de sus principales diferenciadores es la cantidad de datos que registra en tiempo real. De acuerdo con GE, en tan solo un vuelo se genera alrededor de un terabyte de datos. Estos datos son



utilizados por los pilotos para tomar decisiones que impacten la eficiencia, y por las aerolíneas para optimizar sus rutas y anticipar problemas de mantenimiento.

¿Qué habilita a estas empresas para utilizar datos para obtener ventaja competitiva?

Una organización dirigida por datos es aquella que adquiere, procesa y aprovecha datos continuamente para crear eficiencias, desarrollar o iterar nuevos productos, y navegar el panorama competitivo.

Desmenecemos esta definición...

El primer paso para trabajar con datos es poder adquirirlos y procesarlos. Pero no es obvio que es lo que tenemos que hacer para lograr esto de forma continua. Las mejores organizaciones dirigidas por datos se enfocan sin cesar en mantener sus datos limpios. Los deben estar organizados,

documentados, en un formato consistente, y libres de errores. Limpiarlos típicamente representa 80% de un esfuerzo de ciencia de datos. Establecer procesos para limpiar datos a grandes escalas agrega mayor complejidad. Las organizaciones exitosas hacen inversiones significativas en herramientas, procesos y auditorías para maximizar dicho proceso. Han desarrollado una cultura que comprende la importancia de la calidad de los datos.

Un número sorprendente de organizaciones ha invertido cuantiosamente en sus capacidades de procesamiento de datos, con la esperanza de que eventualmente sus empleados generen valor a partir de ellos. Esta actitud de tipo "si lo construyes, eventualmente vendrán" rara vez es exitosa. El resultado son grandes costos para crear un baúl de datos que rara vez se usa. Las mejores organizaciones utilizan sus datos continuamente para entender a sus clientes y los detalles de su negocio. Hacen experimentos que les permiten demostrar hipótesis que mejoran su empresa y procesos; y utilizan los datos para construir nuevos productos.

DEMOCRATIZAR LOS DATOS

La democratización de los datos es una de las ideas más poderosas que ha surgido de la ciencia de datos. Todos los miembros de una organización deberían tener acceso a tantos datos como sea legalmente posible.

El acceso público a datos se ha popularizado en las ciencias (por ejemplo, algunos países dan acceso directo a sus datos del servicio climatológico o de salud). En el caso de las empresas, Facebook es una de las primeras organizaciones que dio acceso general a sus empleados a los datos de la empresa sin necesidad de hacer una petición o esperar autorización. Esta fue una idea radical ya que la noción preexistente era que los empleados no sabrían cómo acceder a los datos y utilizarían datos incorrectos para tomar decisiones de negocio incorrectas. Aunque ciertamente hubo retos que superar, Facebook encontró que los beneficios superaban por mucho a los costos; se convirtió en una empresa más ágil, que rápidamente puede construir nuevos productos y responder a cambios en el mercado. El acceso a datos se ha convertido en una parte crítica del

éxito de Facebook, y es algo en lo que continúa invirtiendo agresivamente.

Las principales empresas de internet han seguido el ejemplo de Facebook. Ser capaz de utilizar SQL para acceder datos es una habilidad esencial para todos los analistas y ejecutivos en empresas como Google y LinkedIn. Las organizaciones sin fines de lucro también están viendo grandes beneficios de brindar acceso a sus datos. Se han dado cuenta que expertos fuera de la organización pueden realizar descubrimientos importantes. Por ejemplo, el Banco Mundial abrió sus datos para que grupos de voluntarios puedan limpiarlos e interpretarlos. Esto ha generado tanto valor que creó un sitio web dedicado a datos públicos.

Los gobiernos también han comenzado a reconocer el valor de democratizar el acceso a datos, tanto a nivel local como nacional. Por ejemplo, el gobierno del Reino Unido ha sido líder en esfuerzos de datos abiertos, y el gobierno de Estados Unidos creó la Open Government Initiative para aprovechar este movimiento. Conforme el público y el gobierno comenzó a ver el valor de abrir los datos, los gobiernos empezaron a catalogar sus datos, capacitar al público sobre cómo usarlos, y publicarlos de manera que sean fácilmente consumibles con herramientas. En la ciudad de Nueva York, el acceso a datos llevó a nuevas estrategias basadas en analítica, que obtuvo grandes resultados, incluyendo una mejora de 5x en el tiempo que le toma a los inspectores encontrar departamentos ilegales.

Uno de los retos de la democratización es el de ayudar a las personas a encontrar los conjuntos de datos adecuados y asegurar que dichos datos son de buena calidad. Como ya mencionamos, 80% del esfuerzo del científico de datos es preparar los datos, y es probable que la mayoría de las personas no cuente con la preparación adecuada para realizar la limpieza ellos mismos. Es así que para ayudar a los empleados a utilizar datos de la mejor manera, en algunas organizaciones está surgiendo el rol del delegado de datos (data steward). Su responsabilidad es asegurar la consistencia y calidad de los datos, y gestionar la inversión en herramientas y procesos para lograr esto. 📊



¿CÓMO PUEDE LA ANALÍTICA PREDICTIVA AYUDAR A MEJORAR LOS NEGOCIOS?

Por Manuel del Pino

● **El descubrimiento de información interesante** siempre es valioso y puede ayudar a realizar análisis con relación a un incidente. Dicho análisis se produce cuando una organización descubre una serie de tendencias sobre comportamientos de sus clientes, como pueden ser los siguientes:

- Clientes que cancelan sus cuentas y se van a la competencia.
- Clientes menores de 30 años que cuentan con historiales reducidos de hábitos de compra.
- Clientes dentro de una horquilla determinada de ingresos que han llamado al call center tres o más veces en los últimos seis meses.
- Clientes desde hace menos de un año cuyos comentarios sobre la empresa reflejan sentimientos negativos.

Sin duda, estas conclusiones resultan interesantes y posibilitan un conocimiento más profundo del comportamiento de los clientes, pero para extraer su máximo valor deben tener impacto directo en el negocio. Para ello han de permitir predecir el comportamiento del cliente en el futuro para poder modificar los procesos y decisiones de la empresa.

Una creciente cantidad de empresas está optimizando sus negocios gracias a la monetización de datos y al empleo de la analítica para tomar mejores decisiones. Aunque estas organizaciones provienen de numerosas y variopintas industrias, en esta ocasión me gustaría destacar dos áreas en particular: en el ámbito universitario, la relativa al reclutamiento y retención de estudiantes; y en el segmento sanitario, la relacionada con la calidad de los datos para mejorar la atención al paciente. Ambos escenarios ejemplifican a la perfección cómo la analítica predictiva puede ayudar no solo a destapar tendencias ocultas en los datos, sino también a generar un impacto en la cuenta de resultados. La habilidad para generar ingresos, reducir costos y mejorar el servicio ilustra el hecho de que obtener ideas muchas veces no es suficiente: la clave estriba en ir más allá.

MÚLTIPLES VARIABLES

Las conclusiones que obtenemos con relación a la edad de un cliente, su historial de gasto, sus ingresos, las llamadas que realiza a nuestro call center, su entorno geográfico o las opiniones que ha vertido sobre nuestra organización nos pueden ayudar a predecir su futura marcha. Podemos asignar una probabilidad de fuga a cada cliente

y realizar una aproximación más proactiva y cercana a aquellos con niveles más altos. Esto permitirá reducir el índice de fuga, ahorrar dinero, mejorar la satisfacción del cliente y mantener cuota de mercado.

Obtenemos conclusiones cuando analizamos los datos de manera holística, combinando, por ejemplo, aquellos que reflejan las interacciones de los clientes con los call centers y las redes sociales, junto a datos tradicionales como cuentas, información de gastos y detalles geográficos. Pero la analítica que realmente impacta en el negocio se obtiene, en este caso, tras vincular esas repetidas llamadas al call center y los sentimientos negativos vertidos en redes sociales con una incidencia concreta, como la fuga a la competencia.

Hoy día ya somos capaces de saber cómo estos aspectos elevan los costos de una organización, cómo pueden interactuar con otras fuentes de información y cómo permiten realizar análisis de las causas y mejorar los procesos. De esta manera cualquier organización podrá establecer acciones correctivas para reducir el número de llamadas de queja y mejorar el *engagement* que su marca despierta en los distintos canales sociales. ☺

CAPITAL DE DATOS Y ANALÍTICA AVANZADA

Por Jorge Gálvez

● **Para ser exitosas en la economía digital**, las empresas deben internalizar la idea de que los datos son un nuevo tipo de capital, tan importante como el capital financiero para el desarrollo de nuevos productos y servicios.

Las implicaciones de esto van mucho más allá de la toma de decisiones por medio de analítica predictiva. Para algunas empresas, el capital de datos incluso sustituye al capital tradicional. Solo de esta manera se puede explicar que empresas de servicios digitales tengan valuaciones de miles de millones de dólares.

Hoy en día, los analíticos avanzados permiten a las empresas sacarle el mayor provecho a los datos que se producen a partir de las actividades del día a día, desde la información generada por sensores, por los puntos de venta, llegando a las redes sociales e imágenes y video. Las herramientas tradicionales de inteligencia de negocio producen un resumen o sumario de los datos, generando a su vez más datos, pero las herramientas tradicionales de inteligencia de negocio proveen una visión del pasado, es decir, lo que ya sucedió. Los analíticos avanzados, de la misma forma, producen más datos a partir de los datos existentes, pero estos datos son transformacionales, creando conocimiento previamente desconocido y ofreciendo una visión del futuro y de opciones no identificadas previamente, que muy posiblemente sucederán.

Las empresas deben de poner a trabajar a sus datos para poder obtener un retorno de inversión a partir de la captura, limpieza y mantenimiento de los mismos. Poner a trabajar a los datos a través de

algoritmos de analíticos avanzados, ha mostrado drásticamente un constante retorno de la inversión. Tradicionalmente, los centros de datos han sido vistos como un impuesto o costo más que un generador de dividendos. El costo de mantener grandes volúmenes de datos es una carga bastante pesada cuando no es posible sacar nuevo valor de los datos y utilizarlos para generar acciones de apoyo al negocio, tales como predecir la conducta de los clientes, entender las causas raíces del comportamiento y reducir los fraudes.

¿Cómo pueden las compañías convertir los centros de datos en activos generadores de dividendos?

Una forma es a través de la reducción del costo de transacción. Típicamente, estos costos de transacción involucran mover los datos en ambientes donde los modelos predictivos pueden producirse o el análisis de los datos se realiza sobre un subconjunto pequeño de datos que puede caber en arquitecturas de hardware y software existentes. Luego está el costo involucrado en poner dichos modelos en producción. Los costos de transacción resultan en esfuerzos de múltiples pasos que involucran un trabajo intensivo, lo cual lleva a las compañías a posponer su inversión en los datos y obtener valor de ellos.

LEYES DEL CAPITAL DE DATOS

En Oracle identificamos tres leyes fundamentales del capital de datos que describo a continuación:

1. Los datos provienen de la actividad. Los datos son registros de eventos sucedidos, así que si no estamos al tanto de las actividades cuando suceden, perdemos para

siempre la oportunidad de capturarlos. Así que las empresas deben digitalizar y “datificar” las actividades clave con clientes y proveedores para poder estar capturando esos datos al momento de la actividad.

2. Los datos generan más datos. Los algoritmos que dirigen las decisiones de precio, publicidad, inventario o detección de fraudes generan datos que a su vez utilizan como retroalimentación para mejorar su desempeño. Este ciclo de retroalimentación genera una ventaja competitiva que es difícil de alcanzar.

3. Las plataformas ganarán. En las industrias intensivas en información, como el software y el mercado bursátil estamos acostumbrados a que existan plataformas informativas, ya sea abiertas o dominadas por una empresa. Este mismo fenómeno se dará en otras industrias conforme usen datos más y más. En industrias como la automotriz y de servicios de salud veremos que las empresas competirán para convertirse en la plataforma de información dominante en su industria.

CONCLUSIÓN

Las organizaciones hoy en día deben considerar al científico de datos o, a los analistas como “gerentes del capital de datos”. Esto es, son las personas encargadas de extraer el mayor provecho y ganancia de los activos de datos. Para alcanzar estos altos dividendos con costos bajos de las transacciones, los gerentes del capital de datos necesitan valerse de herramientas y plataformas que automaticen las actividades, haciéndolo más productivo y finalmente obteniendo logros más espectaculares para la empresa, haciendo más con menos gracias a la rapidez y sencillez. 🌐

TRANSFORMANDO CON DATOS LA INDUSTRIA DE MANUFACTURA

Por Miguel Nhuch

● **Ahora más que nunca**, la industria de la manufactura se enfrenta al reto de reunir y comprender cantidades masivas de datos para impulsar la eficiencia operativa, un nivel más alto de servicio y soporte. Nos guste o no, la manufactura avanza cada vez más rápido y la actitud de conformismo ante la situación actual empieza a desvanecerse porque los encargados de la toma de decisiones ya no pueden contar con los antiguos sistemas de negocios para obtener la información que necesitan.

En el mercado de hoy, explorar el efecto y la interacción entre la eficacia en la producción, la calidad de los productos, la demanda de los clientes y la excelencia en el servicio simplemente no es posible sin análisis significativos. Estas son cuatro de las formas en que los fabricantes están revolucionando su industria por medio de los datos:

1. Mejorar la producción, el desempeño de las plantas y los productos con analítica de autoservicio.
2. Mejorar la planeación de ventas y operaciones por medio de combinación y predicción.
3. Movilizar la cadena de suministro con análisis en tiempo real.
4. Escuchar, interpretar y reaccionar más rápido a los comentarios de los clientes.

MEJORAR EL DESEMPEÑO EN LAS PLANTAS CON ANALÍTICA DE AUTOSERVICIO

Tradicionalmente, las personas que están dentro de la manufactura han obtenido acceso al conocimiento de los datos mediante informes estáticos de aplicaciones empresariales y herramientas de inteligencia empresarial que únicamente administra y usa el departamento de Tecnología de la Información. Esta costumbre antigua, diseñada y desarrollada principalmente durante la década de los 90, generalmente es compleja, inflexible

y toma mucho tiempo. Debido a que las mejores implementaciones de análisis son tableros creados por el usuario, que funcionan sobre una infraestructura administrada por IT, la optimización del auto-servicio es clave.

La analítica de autoservicio permite que los empleados de las plantas de manufactura puedan, al igual que el resto de la organización, visualizar y comprender los datos a lo largo de la cadena de demanda dentro de las operaciones de producción y durante todo el ciclo de vida del servicio. Al añadir visibilidad al desempeño operativo, los empleados tendrán la capacidad de supervisar los datos a lo largo de toda la organización y de aplicarlos para buscar mejoras continuas en los negocios y en los procesos por medio de las filosofías de Seis Sigma o los principios de producción sin desperdicios. El autoservicio también apoya la implementación del marco DMAIC (Define, Measure, Analyze, Improve, Control) para apoyar el ciclo de mejora impulsado por los datos, lo cual permite que un individuo explore e identifique la causa original de los defectos en los productos o los cuellos de botella.

Tesla Motors Designs, un importante fabricante de vehículos eléctricos y componentes de sistemas de transmisión, descubrió que los análisis de autoservicio permitieron a sus empleados explorar sus propios datos y contribuyeron de forma importante a sus descubrimientos acerca de la mejora y la estabilización de la producción. De acuerdo con sus ingenieros de prueba, cuando las personas empiezan a tener acceso a los datos, comienzan a hacer más preguntas; y ahí está la capacidad de ir más allá. Cuando los ingenieros de Tesla están tratando de encontrar la causa original de algún problema molesto que ha estado afectando la producción de forma negativa durante semanas o meses,

el poder revisar los datos y ver cosas que no habían visto antes es una gran ventaja.

La figura 1 muestra un ejemplo de un tablero de análisis de producción donde los usuarios pueden explorar el desempeño de varias órdenes producidas con dos máquinas distintas (123 y 456). A la izquierda vemos las distintas órdenes graficadas de acuerdo a su tamaño y varianza. Al seleccionar una de ellas podemos ver a la derecha los detalles de dicha orden, como el tiempo de configuración y velocidad de operación.

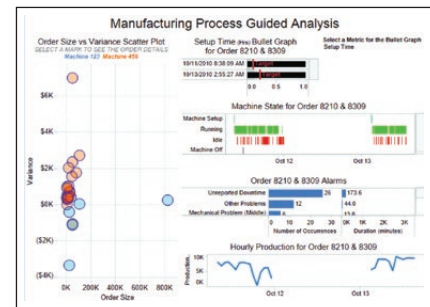


Figura 1. Tablero de análisis de producción.

MEJORAR LA PLANEACIÓN DE VENTAS Y OPERACIONES

Las grandes mejoras en la manufactura deben comenzar desde la fuente: la cadena de suministro. Es esencial que todos los profesionales de cadenas de suministro tengan la capacidad de proporcionar bienes y servicios usando diferentes sistemas de información en plazos muy cortos.

Los fabricantes de hoy se enfrentan a muchas fuentes de datos: la planeación de la fuerza laboral y los pedidos desde el sistema de planeación de recursos empresariales (ERP), la información de los pedidos desde los sistemas de ejecución de manufactura (MES), las bitácoras de tiempo y asistencia, los datos de alarmas y producción de diferentes fabricantes de



equipo y varios sistemas de control de línea de producción (PLC) y de control de sistemas y adquisición de datos (SCADA). Vincular estas islas de información es clave para comprender el panorama y la toma de decisiones.

MOVILIZAR LA CADENA DE SUMINISTRO CON ANÁLISIS EN TIEMPO REAL

La capacidad de ver y comprender lo que sucede con datos en tiempo real y desde un dispositivo móvil es aún más revolucionaria en las operaciones de cadenas de suministro. Los datos de manufactura cambian constantemente, pero son relevantes de manera inmediata. Usar los datos en el momento correcto es esencial para una operación más provechosa.

La buena noticia es que el BI móvil está ofreciendo información en el momento y lugar necesarios para tomar decisiones cruciales de manera rápida. Coca-Cola Bottling Co. Consolidated (CCBCC), la empresa embotelladora independiente


de Coca-Cola más grande de Estados Unidos, tiene trabajadores en el campo que interactúan y colaboran con tableros visuales desde cualquier lugar; incluso los choferes de camiones en diferentes ciudades. CCBCC resolvió un cuello de botella enorme en su cadena de suministro, debido a la disponibilidad limitada de informes, proporcionando actualizaciones diarias del tablero en dispositivos móviles a los líderes y a más de 800 empleados.

REACCIONAR MÁS RÁPIDO A LOS COMENTARIOS DE LOS CLIENTES

Después de todo, los deseos y las necesidades del cliente importan. Los fabricantes tienen que reunir los datos de los clientes mediante muchos canales diferentes, como las redes sociales, los centros de atención telefónica y las encuestas de los clientes. Cuando se llega a comprender algo gracias a la información de los clientes, es fundamental actuar rápido.

Trane, líder global en sistemas y equipo de aire acondicionado, pasó de usar únicamente hojas de cálculo a integrar visualizaciones de datos de autoservicio con los datos de servicio de los clientes para mejorar de manera significativa su velocidad de comprensión. Los tiempos de entrega a partir del entendimiento de los datos hasta la satisfacción del cliente son ahora de 10 a 100 veces menores.

Con la visualización de datos, Trane da seguimiento a las métricas clave para registrar los comentarios de los clientes más rápido. Tienen la capacidad de explorar sus datos para encontrar respuestas, por ejemplo, ¿los técnicos llegan puntuales en una región determinada?

Los empleados de ventas, los empleados de manufactura y los ejecutivos pueden ver rápidamente los distritos con mejor y peor desempeño, así como buscar la razón en los datos subyacentes. 



EL RAZONAMIENTO DETRÁS DEL LAGO DE DATOS EMPRESARIAL

Por Pedro Galván

● **Uno de los conceptos** que ha surgido en los últimos años en el ámbito de gestión de información empresarial es el “lago de datos” (*data lake*). En principio hay quienes podrían pensar que el lago de datos es simplemente una nueva forma de presentar el *data warehouse* y que en realidad no hay nada nuevo bajo el sol. Sin embargo, esto no es correcto.

Es cierto que al igual que un *data warehouse*, un *data lake* es un repositorio de información empresarial. Sin embargo, la diferencia es que un *data warehouse* contiene información altamente estructurada, que se carga de acuerdo a un modelo previamente definido y acordado; es decir, la estructura y requerimientos de información se definen previo a que ésta se cargue (*schema on-write*). Por otro lado, el *data lake* contiene información en su estado nativo, que puede ser estructurada o no estructurada, y el consumidor de la información le da la forma que necesita para un propósito específico cuando la accede (*schema-on-read*); así que un mismo conjunto de datos, que reside en su forma nativa en un lago de datos, puede ser moldeado de distintas formas para atender distintas necesidades. Esto hace que los lagos de datos sean mucho más flexibles y ágiles que los *data warehouse* tradicionales.

En el presente artículo explicamos la necesidad y razonamiento detrás de los lagos de datos.

Peter Drucker utilizaba la expresión “la cultura se come a la estrategia de desayuno” para resaltar que lo más difícil de cambiar en una organización es su cultura. A pesar de esto, en el ámbito de gestión y análisis de información los departamentos de TI tienen décadas luchando contra una cultura de negocio que no pueden cambiar. Mientras los departamentos de TI siguen buscando forzar a las áreas de negocio a utilizar un *data warehouse* centralizado, las áreas de negocio continúan usando hojas de cálculo y bases de datos departamentales.

El concepto del lago de datos busca resolver este desafío por medio del uso de tecnologías de big data (por ejemplo Hadoop) para disminuir las restricciones de costo en almacenamiento y movimiento, al mismo tiempo que provee a las áreas de negocio la flexibilidad y agilidad que requieren.

El lago de datos no es simplemente una iniciativa tecnológica. Consiste principalmente en cambiar la cultura de TI para ser más compatible con la de las áreas de

negocio. Los departamentos de TI deben darse cuenta que no lograrán cumplir sus ambiciones de centralizar bajo su control toda la información de la empresa. El lago de datos atiende este reto por medio de la construcción de una cultura unificada hacia los datos que permita enfocarse en las áreas que generan verdadero valor.

LA CULTURA DE INFORMACIÓN DEL NEGOCIO

Una empresa típicamente está organizada como un conjunto de departamentos, regiones o verticales, cada una con su propia estructura y métricas. La cultura de información en cada una de estas áreas consiste en utilizar vistas locales de información, desplegadas y analizadas en un contexto local.

Incluso a nivel corporativo la intención es tener vistas localizadas y restringidas a los intereses corporativos. Esta cultura fomenta el uso de soluciones puntuales y dificulta la gobernanza horizontal. Si algo le funciona a un área, entonces hay menos motivación para habilitar una vista horizontal.

LA CULTURA DE INFORMACIÓN DE TI

En contraste, los departamentos de TI

históricamente han buscado establecer una solución de tipo “una talla para todos”; es decir, el data warehouse empresarial. La meta es simple: tener toda la información empresarial en un mismo lugar de manera que el modelado carga y gobernanza se haga una vez y ya.

Esta vista centralizada busca dar al negocio lo que sea que necesite y proveer un punto único a donde todos los usuarios puedan ir para obtener información. Esta cultura promueve la centralización con el afán de maximizar el control y la eficiencia de costo.

Esta cultura se resume en unos cuantos principios:

1. Hay una sola vista para cualquier dato: la forma canónica.
2. Sólo se deben mover los datos requeridos.
3. Empujar a todo el negocio hacia la vista única de información.
4. Prevenir bases de datos locales.

La mayoría de los data warehouse empresariales han sido diseñados utilizando una capa central modelada bajo la 3era forma normal (forma canónica). En teoría, esta capa debería ser alimentada por todas las fuentes de datos (internas y externas) y a su vez debería ser la fuente de cualquier *data mart* o mecanismo de explotación de datos.

Esta capa da fuerza a una forma de trabajo de “una versión única de la verdad”. Cualquier información puesta en el warehouse debe definirse de forma estricta y precisa, siguiendo procesos y aprobaciones tanto de áreas de negocio como de TI. Típicamente es un consenso, lo cual implica que nadie obtiene justo lo que quiere y hay un balance de infelicidad.

En organizaciones con una operación sencilla o con mucha homogeneidad entre sus unidades, esto es posible, pero en organizaciones más complejas tendemos a encontrar los siguientes problemas:

- El tiempo requerido para agregar nuevos tipos de datos o actualizar procesos o métricas existentes es demasiado largo como para satisfacer las necesidades del negocio.
- Alinear los procesos de todas las

unidades de negocio es complejo, largo y en algunos casos no deseable.

- Al lidiar con información externa es muy poca la capacidad que tenemos de estandarizar.

Para resolver dicha problemática los departamentos de TI tienden a apoyarse en procesos y herramientas ETL (Extract, Transform, Load). El dilema es que esto es un mecanismo de “mínimo común denominador”, en el cual se extrae la mínima información posible que satisface el esquema y mantiene la consistencia; así es, la prioridad es la consistencia, no el valor a las áreas de negocio.

El razonamiento principal para dicha estrategia es la reducción de costos, ya que al tener una sola vista de la información, TI puede minimizar el costo de administrarla. Sin embargo, dado que esto no resuelve las necesidades de las áreas de negocio, el resultado es que éstas construyen sus propias bases de datos y soluciones para explotarlas. Continúan alimentando al data warehouse corporativo con el mínimo de información acordada, pero en realidad no lo usan como fuente de datos.

NUEVOS DESAFÍOS

El acercamiento previamente descrito está basado en las necesidades de hace 30 años, que para el ritmo al que evoluciona TI es toda una eternidad. Actualmente hay mucho más preguntas alrededor de los datos que requieren respuesta:

- ¿Cómo manejamos los datos no estructurados?
- ¿Cómo enlazamos datos internos y externos?
- ¿Cómo podemos adaptarnos a la velocidad que requiere el negocio?
- ¿Cómo eliminamos el repetitivo proceso de ETL?
- ¿Cómo soportamos distintos niveles de calidad y gobernanza de datos en base a distintas necesidades de negocio?
- ¿Cómo dejamos que las distintas unidades de negocio tomen la iniciativa?
- ¿Cómo aseguramos que la plataforma será adoptada?

Aunado a esto, en los últimos 30 años ha habido un cambio dramático en la

tecnología disponible. Las nuevas tecnologías pueden reducir significativamente el costo de almacenamiento, habilitar analítica en tiempo real y proveer información; y permitir a los usuarios utilizar fuentes de datos diversas.

Son estos desafíos de negocio y avances tecnológicos los que han llevado a un nuevo tipo de solución y metodología: el lago de datos. Estrategia basada en la noción de no estorbar la operación del negocio y fomentar una nueva cultura de información donde se aproveche la cultura de información en las áreas de negocio, en lugar de combatirla.

CONCLUSIÓN

El paradigma del lago de datos se basa en los siguientes principios:

1. Capturar toda la información posible en su estado natural, sin modificaciones.
2. Fomentar que las áreas de negocio construyan soluciones puntuales.
3. Permitir que las áreas de negocio decidan la relación costo/desempeño de su solución.
4. Concentrar la gobernanza solo en los puntos requeridos.
5. Considerar la vista corporativa como si fuera una vista departamental más.
6. La información no estructurada sigue siendo información.
7. Asumir que el lago contiene todo.
8. La escala es dirigida por la demanda que es elástica (se puede expandir o contraer).

Es en estos principios que se basa la nueva forma de gestionar el repositorio de datos empresarial. Es un nuevo paradigma que promete entregar lo mejor de dos mundos: permitir que las áreas de negocio tengan la flexibilidad y riqueza de información que buscan, de una forma integrada y con bajo costo de propiedad. ☺

Referencias

- [1] “The Principles of the Business Data Lake”. Capgemini & Pivotal, 2013



CUATRO RAZONES POR LAS QUE FALLAN LOS PROYECTOS DE ANALÍTICA

Por Paige Roberts

● **Recientemente impartí una conferencia** con un colega en la que mencionó que 60% de los proyectos de big data fracasaban. Inmediatamente los participantes nos preguntaron cuáles eran las principales razones. Conforme más y más organizaciones buscan extraer valor de sus datos y se embarcan en iniciativas de analítica, es importante conocer cuáles son las principales fallas por las que pueden fallar estos proyectos. A continuación comparto las cuatro razones más comunes por las que he encontrado que fallan los proyectos de analítica

CARENCIA DE UN CASO DE NEGOCIO

El principal problema que he encontrado en proyectos de Hadoop en los últimos años, es considerar que implementar una plataforma de analítica es un fin por sí mismo. Uno de mis pasatiempos es el tiro con arco, así que me queda muy claro que para dar en el centro primero necesito tener un objetivo.

Si no entendemos cuál es el dolor de la organización que buscamos remediar, simplemente estaremos haciendo un experimento

Paige Roberts es Product Manager para la línea de Big Data en la empresa Syncsort. Previamente fue consultora en Hortonworks y evangelista técnica en Actian.

sin sentido. Sin un problema en mente, el proyecto no tendrá dirección y por lo tanto el criterio de éxito no estará definido. Ante esto, la probabilidad de que dicho proyecto sea considerado un éxito es prácticamente nula.

EXPECTATIVAS IRREALES

La situación opuesta al problema previamente descrito es ponerse metas irreales, tirarle a las estrellas. La maquinaria de promoción de big data está a todo lo que da, pareciera que el big data puede duplicar nuestras ganancias, ahorrar millones, curar el cáncer, salvar a las ballenas y hasta sacarle esa difícil mancha a nuestra camisa. Ante este tipo de expectativas, cualquier proyecto se quedaría corto. Incluso proyectos que en otro contexto serían considerados exitosos, serían vistos como fracasos si no cumplen todo lo que esperamos del big data.

Existe otra variación de esta situación, y es cuando todas las personas involucradas en el proyecto tienen una expectativa distinta del valor que se obtendrá con el proyecto. En este caso, no hay un objetivo, hay una docena, en direcciones distintas. Tal vez el CMO espera un gran aumento en la efectividad de las campañas de marketing, mientras el CFO espera un gran aumento en ventas, mientras el CIO espera un recorte en costos de infraestructura. Todos esperan que el proyecto impacte su área del negocio. No manejar bien las expectativas tiende a resultar en un "alcance infinito" (scope creep). Si intentamos hacer demasiadas cosas, no lograremos tener éxito en ninguna.

ESCASEZ DE HABILIDADES

Aun si tienes un solo objetivo, y este es claro y alcanzable, no lo lograremos si no tenemos personal con los conocimientos y habilidades adecuadas. Una característica esencial de la analítica de big data es que es algo que no se puede hacer con herramientas y tecnologías tradicionales. Hadoop y su ecosistema de software está madurando rápidamente, pero todavía no es del todo amigable. Una implementación no trivial típicamente requiere integrar por lo menos 20 componentes de software distintos, cada uno con sus particularidades de programación, diseño e integración. Y eso sin tomar en cuenta los detalles de gestionar adecuadamente un clúster de cómputo. Lograr ensamblar este rompecabezas de hardware y software requiere habilidades y experiencia.

FALTA DE CONTINUIDAD

Implementar un proyecto de analítica de big data que sea útil y valioso requiere no solo un objetivo razonable y las habilidades necesarias, sino también tiempo. Por una u otra razón, me he encontrado con que los patrocinadores (champion) de los proyectos de big data dentro de una organización tienden a moverse con frecuencia ya sea dentro o fuera de la empresa antes de que los proyectos lleguen a un estado maduro y que demuestre valor. Digamos que es común que los proyectos de big data queden "huérfanos" antes de ser adultos. Dado que son proyectos complejos, y como ya

mencionamos antes, es probable que no se entienda el objetivo, es poco probable que haya alguna persona o equipo interesada en heredar el proyecto.

¿PERO EN REALIDAD FRACASARON?

Aunque he hablado de las razones más comunes por las que creo que los proyectos de big data fallan, todos los que tenemos experiencia en esto sabemos que en realidad los proyectos no pueden solo clasificarse en éxito rotundo o completo fracaso. Continuando con la analogía del tiro con arco, recordemos que al hacerlo típicamente tiramos varias flechas hacia el mismo objetivo. Difícilmente daremos en el centro la primera vez, pero cada tiro nos sirve para entender y ajustar, buscando que el siguiente tiro sea mejor. Por ejemplo, posiblemente nos demos cuenta que el viento está afectando nuestro tiro y necesitamos compensar.

Recientemente apoyé al equipo de servicios profesionales de Hortonworks en un proyecto bastante grande de implementación de Hadoop. Al integrarme al proyecto, éste se encontraba en una situación complicada y distaba mucho de lograr las metas originales. Platicando con uno de los integrantes del equipo con mayor experiencia, me comentó que en realidad él nunca había visto un proyecto que fracasara por completo. Incluso si no se logran las expectativas irreales de un proyecto, el negocio siempre obtiene algún beneficio. Si no se logra encontrar a la gente adecuada, el personal interno aprenderá y desarrollará nuevas habilidades. Si no tenían una meta clara, experimentarían hasta encontrar algo de utilidad a la empresa. En resumen, esta persona nunca se había encontrado con una empresa que no obtuviera ningún valor de una implementación de Hadoop. Platicando con otros miembros del equipo me encontré con perspectivas similares.

Lo que me hace preguntarme, ¿qué tanto de ese 60% de proyectos fallidos de big data realmente fueron un fracaso?, ¿cuántos fallaron en conseguir su propósito original, pero aún así consiguieron generar valor mayor a su costo? No tengo la respuesta a dicha pregunta, pero si tú estás en una organización que "fracasó" en un proyecto de big data, te recomiendo la consideres.

La versión original de este artículo se encuentra publicada en inglés en <http://bigdatapage.com/four-reasons-why-big-data-analytics-projects-fail-or-do-they/>

Resolviendo Dilemas en un Pipeline de Data Science

Por Blanca Vargas, Andrés Arteaga, y Eduardo Flores

● **Al iniciar un proyecto de Data Science** se tienen que tomar diversas decisiones para que cada etapa del flujo de trabajo o pipeline cumpla con las metas especificadas. El pipeline que definamos depende de factores tales como la experiencia que se tiene con las plataformas, herramientas, lenguajes o algoritmos específicos así como de la investigación previa y referencias externas. Lo ideal es lograr un equilibrio entre la rapidez que nos brinda el uso de los elementos conocidos y la incorporación de herramientas y conocimiento nuevo que permita lograr resultados de forma más rápida y eficiente. En este artículo abordaremos algunos dilemas a los que generalmente nos enfrentamos en la definición de un pipeline y en nuestra experiencia, qué aspectos debemos considerar para tomar la mejor decisión.

Aunque las etapas de un pipeline están muy definidas, todas están ligadas a un objetivo. Y es precisamente la ausencia de un objetivo claro una de las principales causas de los problemas. Para definir ese objetivo debemos cuestionarnos aspectos como: ¿cuál es el resultado tangible esperado del proyecto?, ¿cuáles son los beneficios que generaría el proyecto en caso de ser exitoso?, ¿a quién van dirigidos los resultados? La claridad con la que podamos responder a estas preguntas es fundamental para reducir los contratiempos que invariablemente surgirán.

A lo largo del proyecto nos encontraremos con diversas disyuntivas que nos obligarán a tomar decisiones. Es conveniente reflexionar antes de tomar una decisión impulsiva que haga que ese proyecto fascinante, divertido y retador se transforme en nuestra peor némesis. A continuación se describen algunos dilemas comunes a los que nos hemos enfrentado y que se asocian a las diversas etapas de un pipeline.

ADQUISICIÓN

Cuando inicia un proyecto, uno de los primeros puntos a tratar es, ¿qué información necesitamos para lograr los mejores resultados? En las reuniones iniciales generalmente se hacen acuerdos sobre la información a utilizar y sus respectivas fuentes. Sin

embargo, a medida que transcurren los días no es raro que se presenten las siguientes situaciones:

- Nos damos cuenta de que no tenemos toda la información que se definió en un principio.
- Necesitamos información adicional.
- No tenemos acceso a todas las fuentes.

Es recomendable resolver estas cuestiones lo antes posible. Aunque sabemos que nos encontramos en un proceso iterativo en el cual es posible que hagamos pruebas con datos diferentes a los originales, es mejor tener un conjunto de datos base.

Una situación común es que cuando el proyecto ya se encuentra avanzado se descubren nuevos datos que podrían ser útiles al modelo. Incorporar nuevos datos podría mejorar el resultado pero retrasaría el tiempo de entrega, ¿qué hacer? Esta es una decisión de gestión de proyecto más que nada y podrá variar dependiendo del contexto particular. Sin embargo, nuestra recomendación es guiarse por los lineamientos del desarrollo iterativo, terminando la iteración actual con los datos que se tenían contemplados al inicio de dicha iteración, y evaluar si tiene sentido incorporar los nuevos datos en iteraciones posteriores. Siempre debemos tener en cuenta que al final del proyecto es preferible tener un modelo útil aunque lleve más tiempo, que terminar a tiempo con un modelo inútil.

No tengamos falsas expectativas, el caso ideal en el cual tenemos desde el inicio todos los datos perfectos y completos es solamente una ilusión.

EXPLORACIÓN Y ENTENDIMIENTO

Cuando existen tiempos reducidos podemos tener la tentación de omitir pasos, especialmente en la fase de exploración. A fin de cuentas, confiamos en las fuentes, ¿qué podría pasar? Aun cuando tengamos datos muy limpios, es necesario asegurarse de que no contienen información que nos pueda llevar a resultados erróneos. Por ejemplo, podemos encontrar rangos de edades

irreales y ciudades que no existen, entre otros posibles hallazgos que si no los notamos a tiempo nos harán padecer en fases posteriores. Llevar a cabo la exploración no debe ser un dilema, debe ser un paso riguroso antes de seguir con el proyecto.

Pero explorar los datos no lo es todo. Debemos estar seguros de que entendemos cada variable o atributo. Si contamos con un diccionario de datos la labor se facilita pero si no contamos con él, vale la pena no suponer nada e investigar con el administrador de base de datos de su preferencia sobre el significado de los datos. No hacerlo puede tener un costo elevado.

PRE-PROCESO, TRANSFORMACIÓN, MANIPULACIÓN

Esta fase requiere que tomemos decisiones con respecto a las transformaciones que nos convienen para obtener los mejores resultados. Es necesario conocer bien los datos: ¿de qué tipo son?, ¿qué efecto tienen estos tipos en las transformaciones? Existen detalles sutiles que no deben dejarse para después. No es lo mismo una variable tipo fecha que una variable numérica; no es lo mismo una variable tipo factor que una tipo texto. Aunque visualmente se vean iguales, si no damos a las variables el formato adecuado podemos consumir tiempo valioso.

En cuanto a los valores atípicos (outliers), también nos enfrentamos a dilemas: ¿los quitamos?, ¿los dejamos?, ¿cuáles son las consecuencias de estas decisiones? Como consideraciones generales podemos decir que si los valores atípicos tienen significado, no se deben eliminar. Si por el contrario, valores atípicos distorsionan la realidad entonces deben eliminarse. Imaginemos que estamos obteniendo perfiles de usuario para ciertos productos. Si introducimos edades negativas o mayores de 200 años los modelos resultantes mostrarán patrones de usuarios inexistentes. Por supuesto, hay diversas situaciones y tratamientos que pueden darse a valores atípicos. El punto clave es que no debe restarse importancia a las decisiones sobre estos valores.

ANÁLISIS Y MODELADO

Ya que nuestros datos están preparados y listos, la pregunta es ¿qué tipo de análisis necesito? Esto depende fundamentalmente del objetivo del análisis. Existen infinidad de técnicas y algoritmos que pueden ser útiles ya sea como base o bien podemos diseñar un nuevo algoritmo ad-hoc a nuestro problema. En cualquier caso, la decisión de la técnica o algoritmo a aplicar tiene su origen en el objetivo. Y si no tenemos claro el objetivo es muy difícil esperar tener buenos resultados.

Un problema frecuente es que podemos caer en la tentación de querer aplicar algoritmos o técnicas emergentes y creer que si no lo hacemos no somos competitivos. Por ejemplo, ¿de verdad Deep Learning es la solución a nuestro problema? Tenemos que encontrar argumentos sólidos para tomar decisiones de este tipo. Una estrategia útil es empezar por la solución más simple e ir incrementando el nivel de sofisticación de la solución en la medida que lo requiera el proyecto, no en la medida de nuestro ego.



ROMPEMOS Paradigmas

IMPLEMENTANDO
TECNOLOGÍA AVANZADA

- > UX UI DESIGN
- > WEB & SOFTWARE DEVELOPMENT
- > MOBILE DEVELOPMENT
- > TESTING SOFTWARE
- > TRAINING SERVICES
- > STAFFING AND RECRUITMENT

www.syesoftware.com

info@syesoftware.com



LOS GRANDES DESAFÍOS
REQUIEREN EXPERIENCIA.
**CONOCEMOS EL CAMINO,
PERMÍTENOS ACOMPAÑARTE.**



Nuestros Reconocimientos



www.qualtop.com

info@qualtop.com

COMUNICACIÓN Y OPERACIÓN

Y llegamos a la fase en la que ya tenemos resultados buenos. ¡Si, lo logramos! Después de varias iteraciones en las fases anteriores hemos llegado a la parte de presentar los resultados. La interrogante es: ¿cómo presentarlos?, la respuesta depende de diversos factores: ¿el resultado es un reporte?, ¿es un producto?, ¿es un servicio?, ¿es un dashboard?, ¿es un resultado final o una iteración?, ¿para quién son?

Aunque estemos muy felices con nuestros resultados, una presentación inadecuada puede ser desastrosa si no mostramos los aspectos de interés para cada público. Es conveniente elaborar dos presentaciones base:

- Presentación técnica: debe incluir detalles de análisis e implementación para que sea reproducible. Debe incluir referencias a los repositorios donde se ubica el proyecto para que los interesados puedan repetir los experimentos y análisis. En este tipo de presentación es importante contestar la pregunta: ¿cómo se obtuvieron los resultados?
- Presentación no técnica: debe omitir detalles de análisis e implementación e incluir resultados duros y cómo aplicar esos resultados para lograr el objetivo del negocio. Este objetivo puede traducirse en beneficios económicos, incremento en ventas, campañas de marketing, incremento en número de usuarios, entre otros. La pregunta a responder es: ¿qué beneficio se obtiene con esos resultados?. En este tipo de presentación se debe ser cuidadoso en no utilizar tecnicismos a menos que sea estrictamente necesario. Es nuestra responsabilidad describir las cosas de modo que esté al alcance de profesionales de diversas áreas. Y eso no es una tarea trivial.

DILEMAS TÉCNICOS

La implementación de un producto basado en análisis de datos masivos provoca retos interesantes a resolver dado que los sistemas productivos generalmente no están diseñados para el tipo de carga que implica. Por ejemplo, una base de datos que soporta una aplicación web de alta escalabilidad puede responder adecuadamente cuando se tienen millones de peticiones concurrentes pero cuando se necesita consultar para análisis de datos o para un producto basado en Big Data puede poner en riesgo los sistemas productivos. ¿Qué hacer entonces?, ¿Implementar una solución batch/offline sacrificando posible utilidad de una solución en tiempo real para el usuario?, ¿replicar la base de datos a otra de sólo lectura o con fines exclusivos para el producto de data science?, ¿pagar una nueva instancia en caso de bases de datos comerciales o escoger una alternativa abierta? ¿exportar simplemente a archivos de texto y dejar que los scripts hagan el

trabajo pesado? ¿utilizar cómputo en la nube para el procesamiento, y si es así, cómo hacer segura la información privada que saldrá de nuestros servidores?

Durante la fase de diseño del sistema, uno de los primeros dilemas que se pueden encontrar en las implementaciones es el tipo de arquitectura en la que va a correr. En mayor parte se cuenta con dos opciones: utilizar la infraestructura local o utilizar las diferentes servicios que ofrecen procesamiento en la nube. En este caso es muy importante conocer las ventajas y desventajas que cada una de estas implementaciones ofrece, además de las necesidades propias del sistema a implementar y que pueden afectar en menor o mayor medida a la decisión que vamos a tomar.

Lo primero que tenemos que tomar en cuenta es la capacidad de procesamiento necesaria para la implementación de nuestra solución de data science, de esta manera podemos estimar si los servidores propios son lo suficientemente capaces para soportar la carga extra que vamos a agregar al nivel de trabajo diario. En caso de que el procesamiento no sea factor, se debe de tomar en cuenta la cuestión monetaria ya que las soluciones en la nube involucran un gasto que comienza siendo moderado pero rápidamente puede llegar a cifras altas dependiendo de las instancias que necesitamos crear y el tiempo que las tengamos operando.

Además de las barreras económicas y de procesamiento, existen requerimientos específicos del sistema. En este caso si se elige una implementación de cloud, es necesario prever la integración necesaria con el sistema que tenemos local. Necesitamos saber si es necesaria la replicación de información almacenada en bases de datos locales y que se tendrá que estar replicando de manera constante a la solución en la nube, además de otros aspectos como la cantidad de información que será transferida desde el ambiente local a la nube y viceversa.

Si por otra parte se opta por una implementación local, es necesario prever desde un principio la necesidad futura ya que si es necesario escalar la implementación y no se cuenta con la infraestructura necesaria para hacerlo es posible que se requiera una reimplementación del sistema. Esta decisión puede llevar al consumo de tiempo y recursos que se pudieran haber invertido en otro tipo de solución.

CONCLUSIÓN

Un proyecto de data science representa un reto en distintos niveles y los cuestionamientos son inevitables en el transcurso de las diversas etapas del mismo. Sin embargo, por sobre cualquier aspecto técnico está el factor humano. La colaboración, capacidad, motivación y buena disposición de los miembros del equipo son los factores clave que determinarán si el proyecto tendrá éxito o no. ☺



Administra la agenda del evento, conferencias, ponentes y documentos en tiempo real

Envía alertas y notificaciones a todos tus invitados

Conoce la opinión de tus invitados creando encuestas y votaciones

Comparte itinerarios de vuelos, hoteles y traslados dentro de Eventto App

www.eventto.com.mx

solicita una demostración



Personaliza **Eventto** con los colores y el logotipo de tu marca como algunos de nuestros clientes



contáctanos
info@eventto.org



Cómo Iniciar Tu Carrera en UX Design

Por Misael León

● Aunque el **User Experience (UX)** se ha convertido en un término popular en nuestra industria, siento que todavía hay confusión sobre la verdadera aportación que un UX designer puede hacer a un equipo. Esto ocasiona que diseñadores gráficos y web designers que desean hacer una transición a UX Designers no tengan claro cómo hacerlo. La abundante información que existe en internet termina creando más confusión todavía. ¿Por dónde comenzar?

En este artículo intento esclarecer la confusión y darte algunas recomendaciones básicas para iniciar tu propia carrera en UX.

UN UX DESIGNER ES UN FACILITADOR

Básicamente un UX Designer es el intermediario entre el usuario y la organización. Su responsabilidad es facilitar que todos los involucrados en el desarrollo de producto o servicio entiendan el problema real que se está resolviendo.

Para esto es necesario conocer primero la audiencia, su estilo de vida, sus frustraciones, sus hábitos de tecnología, y el contexto en el que el producto será utilizado. El UX Designer organiza actividades de investigación para generar este conocimiento. Pero no lo hace en solitario, debe incluir al equipo desde el inicio.

El siguiente paso es generar una idea básica y un prototipo inicial, y validarlo con usuarios para asegurar que la solución propuesta es la ideal.

Los prototipos pueden variar en fidelidad desde hechos a papel y lápiz a hasta prototipos de alta fidelidad. El artículo "4 Actividades UX Para Resolver Controversias de Desarrollo"[1] provee una guía para determinar cuándo usar cada nivel de fidelidad y qué herramienta utilizar. Cabe mencionar que el UX Designer no necesariamente produce el diseño final de la interfaz visual, ese es el trabajo de un Visual Designer si lo existe. De otra manera puede generar un estilo básico y los wireframes correspondientes para guiar al equipo a través de la evolución de la solución.

DISEÑA TU PROPIA CARRERA

Todos los UX Designers que conozco hemos pasado por la misma situación. Todos hemos sentido la ansiedad que provoca cuestionar nuestro propio proceso de diseño. De repente ya no es suficiente crear una pieza visual que sea atractiva y que cumpla con los requerimientos que te fueron dados. Uno comienza a hacerse preguntas.

Misael León (@misaello) es UX Design Researcher en Nearsoft, Inc. una empresa de cultura democrática que desarrolla software y produce clientes felices. Es colaborador del UX Clinic, una iniciativa dedicada a difundir las mejores prácticas de UX. Es fanático de los libros, el cine, los chocolates. Promotor de la filosofía del asombro.

Si te está consumiendo la angustia de no saber si tu solución es realmente efectiva, creo que estás listo para hacer la transición. Un UX Designer crea la mejor solución posible pero también levanta los requerimientos, entiende el problema y valida la solución.

Me gusta aplicar esta analogía: Si quieren que prepares la comida, pide que te dejen escoger los ingredientes e ir a comprarlos también.

No tengas miedo a dar el salto. No significa que ya no utilizarás todo lo aprendido en tu carrera de diseñador. Significa que tendrás a tu disposición una serie de metodologías de investigación e ideación de soluciones. Serás como una navaja suiza del diseño. Recuerda que tienes más control sobre tu propia carrera profesional de lo que crees. Es cuestión que te decidas.

Ok, ¿pero cómo hacerlo?

HAZTE VISIBLE EN INTERNET

Comienza a seguir UX Designers nacionales y de otros países. Hay también distintos grupos que constantemente comparten artículos y literatura útil. Aquí algunas referencias:

- UX Booth - <http://www.uxbooth.com>
- UX Matters - <http://www.uxmatters.com>
- UX Magazine - <http://uxmag.com>
- UX Pin - <https://twitter.com/uxpin>
- UX Links - <https://twitter.com/uxlinks>
- UX LLC - <https://twitter.com/ux>
- Usability Tools - <https://twitter.com/UsabilityTools>
- Ideo - <https://twitter.com/ideo>
- User Interface Engineering - <http://www.ue.com>
- Norman Nielsen Group - <http://www.nngroup.com/articles>
- Nearsoft - <http://nearsoft.com/blog/category/design-ux>
- Jared Spool - <https://twitter.com/jmspool>
- Susan Weinschenk - <https://twitter.com/thebrainlady>
- Steven Hooper - <https://twitter.com/shoobe01>
- Laura Klein - <https://twitter.com/lauraklein>

Establece contacto con tantos como puedas. Puedes incluso pedir consejo en algún proyecto actual en el que trabajes. Te aseguro que la mayoría estará gustoso de ayudarte. En realidad todos los UX Designers seguimos aprendiendo y todos hemos pedido ayuda en alguna ocasión. Participa en la comunidad UX, ya sea en eventos locales u online. Si quieres convertirte en uno debes estar donde ellos están. Los eventos de tecnología y de aceleración de startups también son una buena fuente de contactos.

BUSCA UN MENTOR

Reconoce que no eres el experto y que está bien que no lo sepas todo. Es la única manera de seguir aprendiendo. Es altamente recomendable que consigas que alguien te guíe en el camino, un UX Designer experimentado y que esté dispuesto a invertir su tiempo en ti.

El rasgo de personalidad que buscarán en ti es básicamente la curiosidad. No tienes que hacer nada para demostrarla, él o ella se darán cuenta de inmediato.

Este mentor te proveerá literatura útil, autores, libros, pláticas online, referencias, etc. Pero lo hará de manera ordenada dependiendo de tus inquietudes reales.

Descubrirás con el tiempo que el campo de UX es también muy amplio. Ahí creo que radica la parálisis inicial. En la abundante información y no saber por dónde comenzar. Un mentor te ayudará a disminuir esa ansiedad.

SÉ AUTODIDACTA

En una industria que está en constante cambio es primordial estar al tanto de las tendencias. Tus únicas opciones son leer y hacer.

Cinco libros que te ayudarán a poner tu cabeza en el lugar correcto:

1. The Design of Everyday Things, de Don Norman.
2. The Elements of User Experience, de Jesse James Garrett.
3. Change By Design, de Tim Brown.
4. Don't Make Me Think, de Steve Krug.
5. 100 Things Every Designer Needs to Know About People, de Susan Weinschenk.

Al final lo importante es que des rienda suelta a tu curiosidad. Irás descubriendo pistas. Síguelas y escucha a tu mentor.

Es altamente recomendable que lo que vayas aprendiendo en teoría también lo apliques en la práctica. Puede ser en algún proyecto personal en el que estés trabajando actualmente. Por ejemplo, jala a tus conocidos y realiza con ellos una prueba de usabilidad o sal a la calle a platicar con gente que podría utilizar tu producto. Valida tus supuestos con gente real.

Deja tu zona de confort, ahí no sucede nada.

NO TE RINDAS

Es probable que no consigas un trabajo en UX en tu primera aplicación. Todas las organizaciones quieren a alguien que llegue directamente a resolver problemas, para esto necesitas experiencia.

¿Cómo conseguirla si nadie te la oportunidad? Es la pregunta del millón y no tiene respuesta única. Pero eso no puede detenerte. Lee libros, artículos y publicaciones académicas, experimenta con proyectos personales, asiste a eventos, toma entrenamientos, consigue un mentor, entra a comunidades.

La idea básica es que inicies con algo. Inicia hoy.

Cada conocimiento nuevo que adquieras aumentará el valor de tu carrera. Tu futuro es algo que no puedes pasar por alto. ☺

Referencias

- [1] M. León. "4 Actividades UX para Resolver Controversias de Desarrollo. SG Software Guru #49. <http://swqu.ru/rc>
- [2] H. Loranger. "UX Without User Research Is Not UX". Nielsen Norman Group. <http://swqu.ru/rd>

¿Sigue Siendo Importante el Aseguramiento de Calidad?

Por Roselyn C. Piñango

● Las últimas tendencias en tecnología y la dinámica corporativa han incorporado nuevos puntos de control y aseguramiento de calidad incluyendo una gama de herramientas CAST (Computer Aided Software Testing) que facilitan la prevención y detección de defectos. Adicionalmente, las metodologías ágiles incluyen revisiones continuas para evitar el impacto que antes representaba trabajar en islas o encontrar defectos cuando un software ya está en producción. Esto puede llevar a que nos preguntemos: ¿las nuevas tendencias en desarrollo de software pueden reemplazar al aseguramiento de calidad del software (SQA)?, ¿sigue siendo importante el SQA?

La verificación y validación del software son actividades imprescindibles para asegurar la calidad y así, evitar costos por fallas en producción. La calidad no es un valor agregado, es un "commodity" más; la automatización de pruebas parece estar reemplazando a los probadores manuales. En cuanto a las CAST, existen avances en herramientas capaces de derivar casos de prueba en función de algoritmos utilizando técnicas como tablas de decisión.

La prueba de software está muy vigente, pero tiene un enfoque más técnico que antes, predominando los casos de prueba de seguridad y usabilidad en la nube, aplicaciones móviles, Big Data y próximamente en entornos de realidad virtual. Además, el negocio exige pruebas ágiles y las aplicaciones se vuelven más complejas: una aplicación móvil no sólo tiene la misma complejidad funcional que los aplicativos web además cuenta con sus propios requisitos de interfaz, procesos y dependencias de la infraestructura. Las aplicaciones ahora están conectadas, engloban procesos de compra en línea, recolección de datos o trabajan bajo entornos distribuidos.

Ante este escenario, hay un panorama claro para SQA: debe cambiar los paradigmas que han marcado pauta en las acciones de los probadores hasta hoy, para acelerar procesos cuando ahora la velocidad es tan importante como la calidad. A continuación se presentan caminos que son y seguirán siendo tendencia en los próximos años:

Las nuevas metodologías apoyan que el negocio, probadores y programadores formen un equipo multidisciplinario con lo cual deben rediseñarse prácticas arraigadas: cascada, enormes requisitos, incapacidad para probarlo todo, entre otras.

El enfoque de las pruebas debe contemplar verificaciones técnicas, por ello el probador requiere capacitarse en bases de datos, lenguajes de programación, sistemas operativos y entornos que

le permitan indagar en las fallas encontradas en los sistemas o infraestructura asociada a la prueba. No obstante, las empresas siguen valorando a los probadores certificados en estándares como ISTQB, TMMI e IREB porque les permite desarrollar las cualidades analíticas, críticas, minuciosas de un buen probador.

En aras de agilizar los procesos, las herramientas CAST constituyen un elemento fundamental. Existen herramientas para todos los aspectos del proceso de pruebas, así que es importante contemplar herramientas que se comuniquen entre sí, y con otras herramientas utilizadas para otras áreas del ciclo de vida.

Cuando el tiempo y presupuesto se convierten en los "únicos criterios de salida válidos", es el equipo de prueba quien puede tomar las mejores decisiones enfocando las pruebas en base al riesgo.

Las estrategias de pruebas deben ser flexibles y adaptarse a los distintos tipos de pruebas para borrar el mito que el probador es sólo funcional. Se debe impulsar las pruebas de fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

La automatización como mecanismo para agilizar tareas y brindar confianza sigue siendo indispensable sobre todo en la fase de implementación y ejecución para minimizar el número de casos manuales. Igualmente debe considerarse en el nivel de pruebas de componente para que sean utilizados en las pruebas de regresión que se necesiten.

Garantizar el máximo nivel de independencia de las pruebas sigue siendo tendencia y aumenta el número de empresas que contratan fábricas de prueba especializadas para sus procesos de pruebas e impulsar las actividades constructivas y analíticas de SQA enfocadas no solo a detectar sino prevenir los defectos (métricas, estándares, revisiones).

El equipo de SQA se fortalece con nuevas tecnologías que le permiten trabajar de forma remota, ejecutar pruebas aún sin la intervención humana y adueñarse de una visión técnico-funcional difícil de lograr en otros ámbitos del desarrollo de software. Para ello debe adecuar su marco de trabajo con el fin de poder asumir y dar respuesta a las nuevas características y necesidades del mercado. Sin embargo, esto implica una transformación radical en la manera de enfocar las pruebas. Mientras utilice la innovación a su favor en su misión por reducir los riesgos de detectar defectos en producción, brindar información y satisfacer compromisos.

En resumen, SQA seguirá existiendo pues la calidad no caduca. ☺

Roselyn C. Piñango Díaz es Gerente de QA Factory en Global R en Venezuela. Es Ingeniero en Computación con Especialización en Sistemas de Información y tiene más de 10 años de experiencia en desarrollo y pruebas de software.

CTO Forum

¡Únete a los líderes que crean software grandioso!

Comparte ideas, tendencias y experiencias sobre la aplicación de las TI para construir productos y servicios digitales.

Únete a nuestro grupo en:
www.meetup.com/ctos-mx



Próxima reunión:

Mayo de 2016

PRESENTACIONES

CASOS DE ESTUDIO

NETWORKING

DINÁMICAS DE APRENDIZAJE

PLÁTICAS RELÁMPAGO

Contacto:

eventos@sg.com.mx

Resistencia al Cambio y Enfoque en el Cliente

Por Masa K. Maeda

● **Dos clientes con los que hemos trabajado recientemente** decidieron desplegar los siguientes patrones de comportamiento: deseo ferviente de satisfacer al cliente y resistencia al cambio. En principio esto puede parecer muy bueno, pero ¿cuáles son las implicaciones y consecuencias de esos comportamientos en una organización?

El deseo ferviente de satisfacer al cliente parecería ser una forma obvia de incrementar negocio y no estar sujeto a discusión. Sin embargo esto puede ser un gran error y resultar en el colapso de la empresa.

Es correcto enfocarse en el cliente, pero al mismo tiempo es incorrecto hacer todo lo que el cliente solicita. La mayoría de las veces el cliente solamente tienen una idea de lo que desea. ¿Cuántas veces te has reunido con un cliente para que te explique un requerimiento y durante la conversación te das cuenta que tan solo es una idea y la va definiendo sobre la marcha? ¿Cuántas veces conforme construimos lo que solicitó el cliente se generan cambios, adiciones o eliminaciones? Muchas sin lugar a duda.

¿A qué se debe esto? No es que el cliente fue irresponsable o que no dedicó el tiempo suficiente para pensar bien lo deseado. Esto se debe a que siempre se iniciará con una idea e independientemente de cuánto el cliente pudo llegar a detallar, es nuestra responsabilidad ayudar al cliente a madurar esa idea. También se debe a que la naturaleza misma de trabajo de conocimiento es de cambio. Los cambios pueden ser debido a factores externos (económicos, sociales, de mercado) o internos (presupuesto, cambio de estrategia).

Muchos líderes de proyecto piensan que no deben cuestionar al cliente y mucho menos confrontarlo. Estoy de acuerdo en manejar con delicadeza al cliente, pero eso no le resuelve su problema. Si presentamos un argumento económico que muestre el impacto negativo de una idea o solicitud, el cliente lo apreciará y estará aún más agradecido si se le presenta por lo menos una alternativa.

El enfoque al cliente debe hacerse de manera adecuada, pues hacerlo de manera ciega puede fácilmente comprometer a tu

organización y resultar en pérdidas o inclusive en colapso total. Un verdadero enfoque en el cliente sucede cuando en lugar de decir “sí” a todo lo solicitado aseguramos que el cliente reciba lo que realmente necesita.

Con respecto a “resistencia al cambio”, quienes se comportan de esa forma típicamente lo hacen para minimizar riesgo o por creer que sus soluciones ya son las ideales y no hay espacio de mejorar.

En el ambiente profesional y en toda industria, el riesgo más alto es no tomar riesgos porque la competencia tomará la delantera y la empresa misma terminará en la obsolescencia. Esto es aplicable a todo nivel de la empresa, ya sea una división, departamento, proyecto, equipo o individuo.

El temor a fallar puede ser interno, en cuyo caso se debe cuestionar si la persona es apta para ser líder. Pero puede ser también un problema de cultura organizacional como sucede en algunas organizaciones donde las fallas tienen consecuencias muy altas por lo que la gente no se atreve a hacer cosas nuevas o diferentes. Toda empresa, organización, y persona tienen siempre espacio para mejorar. Las personas que piensan lo opuesto están cegadas a la realidad.

Para algunas personas es simplemente conveniente resistir al cambio ya sea para evitar el esfuerzo requerido, por motivo de capital político o financiero. También hay quienes se resisten al cambio simplemente por ser desconocido y no les interesa conocer otras opciones.

Dentro del agilismo se discute mucho todo esto. Sin embargo, no es un reto único del agilismo. En una medida u otra ocurre en todo tipo de empresas, ya sea con respecto a las metodologías ágiles o cualquier otra cosa. Suele ser más notorio en el caso del agilismo porque su naturaleza es más distante a los enfoques tradicionales.

El agilismo no es una moda porque no es algo nuevo (se inició a principios de los 90) y existen miles de casos en todo el mundo que demuestran sus beneficios. Al mismo tiempo es de suma importancia que los tomadores de decisiones se informen bien antes de decidir cuál o cuáles de las metodologías ágiles adoptar. ☺

El Dr. Masa Maeda es CEO fundador de ValueInnova, LLC. Cuenta con clientes de todo tamaño, desde Fortune 500 hasta startups en 12 países en 4 continentes. Es el autor de libro Serious LeAP, y coautor del libro España Lean Startup Nation. Tiene más de 60 publicaciones y numerosas presentaciones magistrales en conferencias internacionales. Previamente hizo investigación y desarrollo para Apple Inc. y fue miembro de los grupos fundadores de 4 empresas en Silicon Valley.

IT Executive Education + CALENDARIO 2016 Entrenamiento Ejecutivo



Febrero 24 y 25, 2016 - Taller

Organizational Change: Lead, Follow, or Get Out of the Way.
Con Sheila Cox



Abril 13 y 14, 2016 - Taller

Leadership.
Con Christopher Avery



Mayo 25 y 26, 2016 - Taller

Business Architecture.
Con William M. Ulrich



Junio 29 y 30, 2016 - Taller

Successful Strategies for Outsourcing in the Cloud.
Con Sara Cullen



Julio 28, 2016 - Taller

IT Strategy.



Septiembre 8 al 10, 2016 - Ciudad de México

Cutter Agile Conference 2016.



Noviembre 14 y 15, 2016 - Cambridge, MA. EUA.

Cutter Summit 2016 - EXECUTIVE EDUCATION+

Nota del editor: En el número anterior de SG se publicó este artículo pero por un error en la edición no se hizo el uso adecuado de índices en las expresiones matemáticas y por ello estas eran incomprensibles. Así que decidimos republicar el artículo corregido. En la siguiente edición de SG continuaremos con la parte 3.

Special Purpose Languages

PARTE 2. ALGUNAS DEFINICIONES

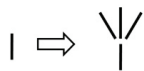
Por Luis Vinicio León Carrillo

● **En el número anterior** hice alusión a la plática que ofrecí en el pasado SG Conference & Expo. Durante ella analizamos brevemente la problemática que dio origen a la llamada “Crisis del software” (a partir de la cual se acuñó por cierto el término “Ingeniería de Software” en los 60), y describimos algunos enfoques que se han aplicado para abatirla, entre ellos los métodos formales, los cuales definimos y describimos brevemente en el número anterior y vimos que utilizan lenguajes formales (que definimos de manera no-formal) procesados por compiladores.

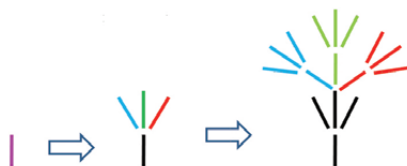
El “estado del arte” de nuestra industria ha estado fuertemente influenciado por el desarrollo de los que llamo: “lenguajes de computación”, que incluyen tanto los lenguajes de programación (de bajo y alto nivel; de primera, segunda, tercera, cuarta y quinta generación; procedurales, funcionales y lógicos, etc.), como los lenguajes de documentación, de especificación, y de arquitectura, entre otros.

En general, dichos lenguajes nos sirven para formalizar patrones que posibilitan la automatización.

Un caso un tanto distinto es el que comenzamos a revisar en el número pasado, en el que les pedí aplicar 3 veces la siguiente regla/patrón haciendo las sustituciones siempre en paralelo.



Por cuestión de espacio expongo aquí solo las primeras 2 transformaciones, en las que los colores tienen la intención de ayudar al seguimiento de las sustituciones (noten que hay ligeras variaciones en algunas inclinaciones de las figuras; finalmente las plantas son sistemas adaptativos).



Como pueden observar, este pequeño patrón, descrito con esa sencilla y única regla, parece describir el crecimiento de una planta (solo en 2 dimensiones). Lenguajes en los que las reglas se aplican en paralelo, se han utilizado para describir fenómenos de este tipo, y han dado lugar a toda una jerarquía de lenguajes conocida como Sistemas Lindenmayer (L-Systems), los cuales a su vez tienen relación con los fractales (el todo contenido en cada parte). Este enfoque viene de la biología (su creadora, Aristid Lindenmayer, era bióloga).

Por otro lado, los lenguajes que utilizamos en la computación utilizan reglas que se aplican no tanto en paralelo, sino más bien secuencialmente, y se definen con un marco conceptual que proviene de la Lingüística (Chomsky es lingüista).

Pero antes de entrar más en detalle con este último enfoque, déjenme abordar algunas cuestiones fundamentales utilizando el enfoque de conjuntos. Por favor, no pierdan de vista que estamos haciendo esto porque queremos describir cómo construir *special purpose languages* propietarios para incrementar la productividad y la calidad de nuestros procesos de desarrollo (y prueba) de software.

Definiciones

Un alfabeto es un conjunto finito de letras. En el caso del Español, ese conjunto tiene 30 letras (contando la ch, ll y la ü), con las cuales podemos construir palabras, oraciones y textos “correctos” en ese idioma, pero también frases que no se consideran parte del mismo (como “añu is morgen”). Podemos decir entonces que el Español es un conjunto de frases consideradas “correctas”, el cual a su vez es un subconjunto del de todas las frases que pueden escribirse con su alfabeto.

Generalicemos un poco y definamos el alfabeto $B = \{b_1, b_2, \dots, b_n\}$, un conjunto finito que contiene n caracteres. Diremos que tiene una cardinalidad de n , y lo escribiremos $|B| = n$.

Definamos ahora la concatenación entre caracteres de un alfabeto B como sigue:

Los sistemas Lindenmayer describen fenómenos en los que las reglas se aplican en paralelo.

$$\begin{aligned}
 b_i \cdot b_j &= b_i b_j \\
 b_i \cdot \lambda &= \lambda \cdot b_i = b_i \quad (\lambda \text{ es la "cadena vacía: una cadena sin letras"}) \\
 (b_i \cdot b_j) \cdot b_k &= b_i \cdot (b_j \cdot b_k) = b_i b_j b_k
 \end{aligned}$$

Adicionalmente, definamos el alfabeto $C = \{c_1, c_2, \dots, c_m\}$; C es también finito, con $|C| = m$.

Definamos la concatenación entre conjuntos de caracteres:

$$\begin{aligned}
 B \cdot C &= \{ b_1 c_1, b_1 c_2, \dots, b_1 c_m, \\
 &\quad b_2 c_1, b_2 c_2, \dots, b_2 c_m, \\
 &\quad \dots, \\
 &\quad b_n c_1, b_n c_2, \dots, b_n c_m \}
 \end{aligned}$$

$B \cdot C$ es también un conjunto finito, con cardinalidad $|B \cdot C| = n \cdot m$.

Definamos ahora la exponenciación de la concatenación aplicada a un conjunto de caracteres:

$$\begin{aligned}
 B^x &= B \cdot B^{x-1} \\
 B^0 &= \{ \lambda \} \quad (\text{El conjunto que tiene un solo elemento, que es la cadena vacía.})
 \end{aligned}$$

B^x es un conjunto finito, de cardinalidad $|B^x| = |B|^x + 1$ (por la cadena vacía).

Ahora definamos la operación conocida como Cerradura de Kleene:

$$\begin{aligned}
 B^* &= \bigcup_{i=0}^{\infty} B^i \\
 &= B^0 \cup B^1 \cup B^2 \cup B^3 \cup \dots \\
 &= \{ \lambda, \quad (1 \text{ cadena de tamaño } 0) \\
 &\quad b_1, b_2, \dots, b_n, \quad (n \text{ cadenas de tamaño } 1) \\
 &\quad b_1 b_1, b_1 b_2, \dots, b_1 b_n, \quad (n^2 \text{ cadenas de tamaño } 2), \\
 &\quad b_2 b_1, b_2 b_2, \dots, b_2 b_n, \\
 &\quad \dots \\
 &\quad b_n b_1, b_n b_2, \dots, b_n b_n, \\
 &\quad \dots \} \quad (n^3 \text{ cadenas de tamaño } 3 \text{ y así sucesivamente})
 \end{aligned}$$

Ciertamente, B^* es un conjunto infinito, pero de los que llamamos "contable", pues podemos utilizar los números naturales \mathbb{N} ($\mathbb{N} = \{0, 1, 2, \dots\}$) para listar todos sus elementos en orden: para $i = 0$ (un número natural) sabemos que hay 1 cadena (otro número natural), para $i = 1$ sabemos que hay n cadenas, para $i = 2$ hay n^2 , y así sucesivamente. Esto permite decir que la cardinalidad de B^* es la misma que la de \mathbb{N} . En otras palabras: dado el alfabeto finito B , podemos construir tantas cadenas con sus caracteres como números naturales hay.

Lo anterior es cierto independientemente de la cantidad de elementos de B . Si B fuera la más grande extensión del código ASCII, esto implica que la cantidad de cadenas de caracteres que podemos escribir en una computadora (o en cualquier dispositivo para procesar información) es infinita, pero es contable.

Finalmente: dado el conjunto A , $\wp(A)$ ("el conjunto potencia del conjunto A ") denota el conjunto conformado por los subconjuntos de A . Algunos ejemplos:

$$\begin{aligned}
 \text{Si } A &= \{ \}, & \text{entonces } \wp(A) &= \{ \{ \}, & & \text{y } |\wp(A)| = 1. \\
 \text{Si } A &= \{ a_1 \}, & \text{entonces } \wp(A) &= \{ \{ \}, \{ a_1 \}, & & \text{y } |\wp(A)| = 2. \\
 \text{Si } A &= \{ a_1, a_2 \}, & \text{entonces } \wp(A) &= \{ \{ \}, \{ a_1 \}, \{ a_2 \}, \{ a_1, a_2 \}, & & \text{y } |\wp(A)| = 4.
 \end{aligned}$$

Podemos demostrar que en general, $|\wp(A)| = 2^{|A|}$.

Ahora estamos en posición de definir con mucha precisión qué es un lenguaje en términos de conjuntos:

Dado un alfabeto finito B , un lenguaje formal sobre B es cualquier subconjunto de B^* (dado un código ASCII extendido de 256 caracteres, Java es un subconjunto especial de cadenas escritas utilizando ese ASCII).

Preguntas

Regresaremos al punto anterior más adelante, pero antes, quisiera hacer notar que a partir de lo descrito anteriormente se desprenden varias preguntas interesantes:

- Dado un alfabeto B , ¿cuántos lenguajes podemos construir con sus caracteres?
- ¿Podríamos procesarlos todos? (v.gr. mediante compiladores o intérpretes)
- ¿Qué tan complejo y eficiente sería ese procesamiento? (en términos computacionales)

Piensen sus respuestas; continuaremos abordando este tema en el siguiente número. ☺

Referencias

[1] L. León. "Los Special Purpose Languages, parte 1". Revista Software Guru #48. <http://sg.com.mx/revista/48/los-special-purpose-languages-0>

Sobre los pasos de Donald Knuth

CONVIRTIENDO AL ARTE EN CIENCIA

Por Gunnar Wolf



Gunnar Wolf es administrador de sistemas para el Instituto de Investigaciones Económicas de la UNAM y desarrollador del proyecto Debian GNU/Linux.

<http://gwolf.org>

La ciencia es el conocimiento que comprendemos tan bien que podemos enseñárselo incluso a una computadora; si no entendemos cabalmente algo, manejarlo es un arte

(...)

De tal suerte, debemos intentar continuamente transformar toda arte en una ciencia: Al hacerlo, lograremos su avance como arte.

– Donald Knuth, discurso de aceptación del Premio Turing (1974)

● **En esta ocasión**, dedicaré mi columna a una persona: Donald Erwin Knuth, alrededor de quien mi atención ha estado volando a últimas fechas, y a quien debo parte importante de mi elección de carrera y desarrollo profesional.

Tenemos el privilegio de trabajar en un campo suficientemente joven como para ser contemporáneos con quienes establecieron su mismísimo significado. Gente que viene de formarse en campos disciplinarios disímiles y fue convergiendo en el naciente campo de la computación, sea por la vía de las ciencias duras, la ingeniería, la administración o el arte.

El árbol genealógico disciplinar de la computación está fuertemente dominado, claro, por los matemáticos. En muchas universidades, las principales carreras relacionadas siguen siendo parte de los departamentos de matemáticas, o (en el caso de las ingenierías) requieren un estudio de un par de años de principios matemáticos apenas para “calentar motores” e iniciar con las materias específicas del ramo.

La computación también se caracteriza por ser muy permeable a gente que desarrolló sus competencias profesionales de manera no formal. ¿Quién de ustedes no ha tenido el gusto de trabajar con colegas cuyo currículum formal parecía apuntar hacia futuros muy distintos? En lo particular, mis colegas y maestros incluyen químicos, músicos, físicos, psicólogos, matemáticos, sociólogos, y un largo etcétera (donde sin duda entramos los autodidactas puros, quienes no seguimos una carrera universitaria formal). Todos ellos comparten un punto en su historia profesional personal: reconocer la ventaja de automatizar procesos

arbitrarios de una forma repetible, controlable, monitoreable. Esto es, darse cuenta cómo para prácticamente cualquier ámbito profesional, puede ser útil aprender a programar (y posteriormente, claro está, entrar al interminable laberinto fractal que supone afinar nuestras habilidades como programador).

Pero muy poca gente puede compararse siquiera con Donald Knuth. Él es uno de los grandes pioneros de prácticamente todas las disciplinas relacionadas con la computación, una suerte de Da Vinci de los siglos XX y XXI.

Y tal como Da Vinci, Knuth probablemente deba ser visto como artista en primer término, dado que todo lo demás es mera consecuencia.

Hemos visto que la programación de computadoras es un arte, puesto que aplica el conocimiento acumulado al mundo, porque requiere de habilidad y de ingenuidad, y especialmente porque puede producir objetos de gran belleza. Un programador que subconscientemente se ve a sí mismo como artista disfrutará de lo que hace y lo hará mejor.

– Donald Knuth, discurso de aceptación del Premio Turing (1974)

El trabajo de Knuth parte de un disfrute estético, no únicamente funcional. Un enamoramiento con los diferentes niveles de lo que su código alcanza. Y eso explica, a distintos niveles de abstracción, el camino por el que llevó a su carrera. Citemos algunos ejemplos.

En 1962, Knuth comenzó a escribir lo que esperaba que fuera un simple libro de texto para la enseñanza de algoritmos —*The art of computer programming*. Cuatro años más tarde, siendo aún un alumno de posgrado, entregó al editor un primer borrador: más de 3,000 páginas manuscritas. Pero siendo un perfeccionista, sería este libro más lo que guiaría prácticamente la totalidad de su vida académica: pasados más de 50 años, el libro (convertido en una serie de volúmenes de referencia) sigue en desarrollo.

Knuth siguió desarrollando y embelleciendo su texto, preparándolo para la impresión. La ancestral técnica de impresión por tipos móviles, que incluía el fundido

de piezas especiales para generar las complejas fórmulas requeridas por las matemáticas, comenzó a encarecer y quedar fuera del alcance para su proyecto.

Conforme avanzaba la tecnología, resultó natural que el libro sería formado por computadora... Pero las herramientas disponibles a principios y mediados de 1970 sencillamente no estaban a la altura estética de una obra con un alta carga de símbolos matemáticos, estructurada con distintos niveles de seccionamiento, cargadas de referencias internas y externas, y Knuth hizo lo que solo un verdadero hacker haría: diseñar su propio sistema de edición de texto, TeX.¹ Liberó la primera versión en 1978, y la última versión mayor que introducía cambios no-compatibles, 3.0, en 1989.

Al día de hoy, Knuth ha terminado formalmente tres volúmenes (de siete que, estima, cubren suficientemente bien el campo) y avanzado fuertemente en el cuarto, que va siendo publicado por fascículos.

Me permito mencionar, como un punto muy lateral y con un dejo de soberbia en el tono, que TeX es uno de los puntos definitorios de mi vida: es el primer programa de cómputo que aprendí a usar. Knuth visitó el IIMAS (UNAM), donde con su ayuda se instaló una versión temprana de TeX. Y al día de hoy agradezco a diario a mi padre, investigador en física, las muchas noches de viernes y sábado que me llevó, aún niño, a "jugar" a la sala de terminales, donde hice mis pininos en TeX en 1983. Para 1985, mi padre y un grupo de colegas editaron empleando dicho sistema el Manual de Lenguaje y Tipografía Científica en Castellano,² libro que recomiendo a todo interesado en la escritura técnica formal y en el correcto uso de nuestro idioma. Hasta donde sé, es la tercera instalación de TeX en el mundo. Si comparan el libro referido con prácticamente cualquier obra impresa en esos años notarán, a pesar de sus imperfecciones técnicas en comparación de nuestra era de super-altas resoluciones, una enorme diferencia cualitativa.

Knuth se comprometió a que, después de la versión 3.0, la interfaz de TeX no tendría ningún cambio que rompiera la compatibilidad; todo texto debería resultar idéntico sin importar en qué equipo de cómputo se preparara. Como un guiño de humor hacker, todas las versiones menores subsecuentes se irían acercando a la perfección simbolizada por el número ①; la versión actual (liberada en enero de 2014) es la 3.14159265. Knuth, quien hoy tiene 78 años y aunque formalmente retirado sigue trabajando para terminar su principal obra a la cual volveremos, autorizó que post-mortem se efectúe un último cambio a TeX, congelando para siempre su versión en ①.

TeX, de la mano del conjunto de macros LaTeX diseñado por Leslie Lamport principalmente entre 1985 y 1995, se convirtió en el sistema de elección para la edición de textos científicos; prácticamente todas

las revistas académicas de las ciencias exactas e ingenierías lo emplean como su principal formato para la entrega de originales.

Cambiemos nuestra actitud tradicional respecto a la construcción de los programas: en vez de imaginar que nuestra tarea principal es darle las instrucciones de qué hacer a una computadora, concentrémonos en explicarle a los humanos lo que queremos que la computadora haga.

– Donald Knuth, *Literate programming*, 1984

Mencionábamos que Knuth apunta a un disfrute estético a diferentes niveles de abstracción. Agreguemos otro nivel más: el paradigma de programación que impulsa; es decir, la computación literaria.

Este paradigma propone que un programa debe ser escrito cual si fuera (dependiendo de su complejidad) como un libro, tratado o artículo estructurado para su lectura humana, en el cual el código fuera el material que acompaña a la exposición lógica. Por medio de un preprocesador, el documento maestro puede exportarse tanto a una versión imprimible como a una compilable. Pueden consultar un sistema de preparación de documentos para documentación literaria en la página de CWEB3 si bien es cierto que no todo programa puede verse beneficiado por esta lógica, la claridad resultante es notoria.

Knuth es, pues, una inspiración en lo estético y en lo funcional, en el purismo matemático y en la limpieza de la implementación. Si acaso, podría criticársele la falta de pragmatismo ante su obsesión por la perfección; sus elevados estándares harían enfurecer a cualquier project manager. Pero sus contribuciones al avance de nuestra disciplina han sido, naturalmente, correspondientemente brillantes.

Espero, con este breve y desordenado texto, dar a conocer algo más de la obra de Knuth, para que inspire a nuestros lectores. Autores de su talla conjuntan ciencia y arte. El tezón para realizar proyectos de décadas de envergadura hace posible avances que otros descartarían por impracticables. Y la apreciación por la belleza lleva a un disfrute en el ejercicio diario de nuestras tareas.

¡Larga vida al maestro Knuth! 🍷

¹ Pronunciado "Tej", Del griego τεχνή, palabra que denota ya sea al arte o a la técnica. La grafía oficial de TeX es con sus tres letras en mayúscula, pero con la "e" más baja, y el espaciado intercaracteres reducido.

² Disponible en <http://www.fis.unam.mx/~bwolf/manual.html>

³ <http://www-cs-faculty.stanford.edu/~uno/cweb.html>

Dinámica de Teclado

NO ES LO QUE ESCRIBES, SINO CÓMO LO ESCRIBES ...

Por Manuel López Michelone

● **Imaginemos que alguien tiene la contraseña** de nuestra computadora y se dispone a ingresar a ella con algún fin dañino. Pero al introducir la contraseña, el sistema no lo reconoce y por ende no le da acceso. ¿Magia? De ninguna manera: es la biométrica del teclado.

La biométrica es la medida de las características físicas y de comportamiento que hace único a cada ser humano. A diferencia de los nombres/contraseñas de los sistemas de cómputo, ciertas medidas biométricas no se pierden, no pueden ser duplicadas o robadas. Pertenecen a cada ser humano en particular. Esto da la posibilidad de sistemas que sean imposibles de ser violados por terceros de manera ilegal, puesto que la información biométrica de los usuarios es única e irrepetible. Las características fisiológicas como pueden ser las huellas digitales, son buenas candidatas para la identificación de personas. La biométrica está ganando popularidad, porque cuando es usada en conjunto con métodos como contraseñas, proveen un nivel extra de seguridad. Algunos de los sistemas biométricos usados para identificación humana se basan en la geometría de las manos, patrones térmicos en el rostro, las venas en la retina, voz y firmas manuscritas. Algunos esquemas son muy invasivos, como leer el iris ocular. La dinámica del teclado es un esquema biométrico poco invasivo y de bajo costo que puede ayudar a complementar otros mecanismos de seguridad.

La manera en cómo los seres humanos escribimos en un teclado de computadora tiene un ritmo específico para cada persona. Es en cierta forma una medición biométrica, pues depende de los rasgos del comportamiento particular de cada usuario. Si podemos medir el tiempo que un usuario tarda en registrar su contraseña en el sistema de cómputo, podremos crear un "perfil" digital que describa este ritmo en su manera de teclear, de manera tal que pueda añadirse como una medida extra a la seguridad de los sistemas.

Hacer un programa que mida los tiempos entre teclas presionadas no es muy complicado. Por ejemplo, en Delphi (lenguaje usado para los ejemplos de este artículo) existen diversos componentes, que permiten medir eventos hasta con una resolución

de un microsegundo. Una opción es el componente THPCounter creado como Mats Asplund, que expone los métodos Start y Read para hacer mediciones de tiempo en microsegundos. El listado 1 muestra cómo podríamos usar este componente para nuestro propósito.

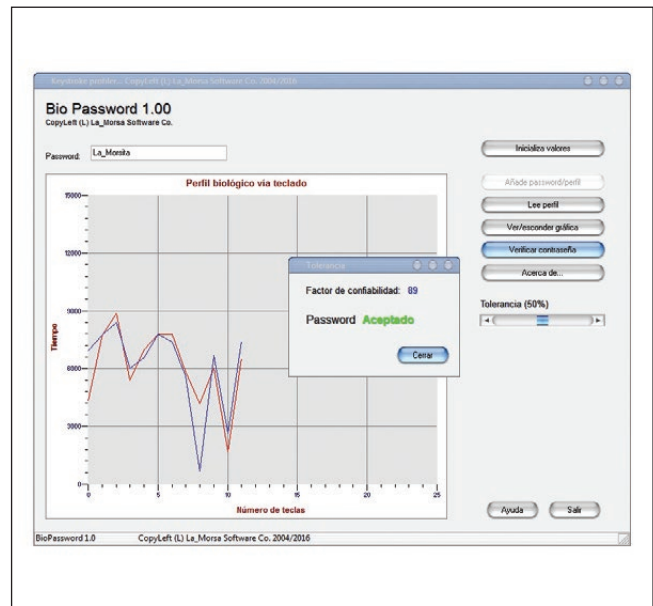


Figura 1. Generación de perfil en base a dinámica de teclado.

Para poder graficar estos datos podemos usar otro componente de código abierto y gratuito: "XYGraph", de U. U.Jürß. Con dicho componente, es trivial generar una gráfica donde en el eje de las X representemos cada presión de una tecla, y en el de las Y el tiempo transcurrido.

Una vez resueltos estos problemas, nació un pequeño programa de demostración que muestra la bondad de la idea. Para que sea más ilustrativo el asunto, una gráfica en el software muestra el perfil biométrico del usuario, que en principio, debería repetirse. Y efectivamente este perfil se repite, no estrictamente con una precisión única, pero sí en promedio. De alguna manera se parece

Manuel López Michelone (@morsa) es Físico por la UNAM y maestro en Ciencias por la Universidad de Essex en el tema de Inteligencia Artificial. Ha sido columnista por muchos años en publicaciones de la industria del cómputo y ávido programador. Actualmente realiza su doctorado en ciencias de la computación en la UNAM. morsa@la-morsa.com

La manera en cómo los seres humanos escribimos en un teclado de computadora tiene un ritmo específico para cada persona.

a la firma que estampamos en un cheque. Es claro que la firma no siempre nos sale estrictamente igual, pero el cajero, al ver nuestro cheque, hace una comparación visual de la firma contra la que tiene en sus registros y decide si paga un cheque o lo rechaza.

De la misma manera, el perfil digital de un usuario al escribir su contraseña puede ser usado para dar acceso a un sistema de cómputo o bien, para rechazarlo. La pregunta en este caso era ¿cómo hacer para comparar el perfil digital contra la contraseña escrita? Existen multitud de ideas de cómo analizar esta información, desde estadística descriptiva hasta el uso de redes neuronales. El enfoque usado fue el de intentar hacer una comparación “visual” entre el perfil de una contraseña y el perfil capturado al alimentar la contraseña para dar acceso al sistema.

Evidentemente la computadora no tiene ojos, así que la alternativa es utilizar los conceptos de cálculo elemental, y así resolver el problema, por lo menos en una primera aproximación. Para mantener simple el problema, se usó el concepto de pendiente. Un perfil está formado por puntos que se conectan (el primero con el segundo, el segundo con el tercero, etc.). Cuando el usuario escribe su contraseña, encontramos que la conexión de estos puntos nos da gráficas de “dientes de sierra”, ya que el brinco de un carácter a otro es discreto (lo contrario de la continuidad). Así, lo que tenemos son dientes de sierra que no necesariamente son todos de la misma distancia. Ésta dependerá del tiempo que se tarde el usuario en oprimir una tecla en particular. De esta manera podemos medir la pendiente, es decir, el grado de inclinación de cada recta que forma el perfil digital.

De la misma manera, cuando un usuario define su contraseña, podemos medir su perfil y guardar esa huella, para compararla posteriormente con el perfil generado cuando alguien intenta ingresar al sistema usando la contraseña. De acuerdo a la cantidad de datos similares, podemos decidir dar acceso o rechazar al usuario. Obviamente, considerando que esta medida es similar a la firma manuscrita, entonces debemos tener un factor de tolerancia en la similitud de las curvas, un umbral, un mecanismo de tolerancia pues es claro que es muy difícil que podamos repetir estrictamente el perfil guardado en la base de datos con el cual se compara cuando escribimos la contraseña. De acuerdo a esto, podemos tener un criterio más o menos flexible, dependiendo del nivel de seguridad que deseemos imponer en nuestros sistemas.

El listado 2 tiene el código que hace la comparación entre una contraseña alimentada y la que el usuario escribe para así hacer la comparación:

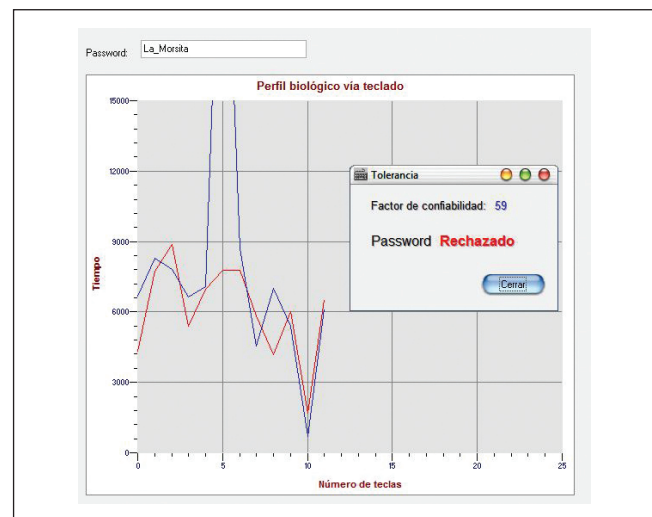


Figura 2. Comparación del perfil obtenido contra el de referencia.

SlopePswd y SlopePerfil son dos arreglos de variables booleanas que contienen la descripción de cuántas pendientes son positivas o negativas.

EJEMPLO REAL

La figura 1 muestra la graficación de dos perfiles generados al escribir la contraseña “La_Morsita” por el mismo usuario. En rojo apreciamos el perfil generado cuando el usuario definió la contraseña originalmente, y en azul el perfil cuando escribió la contraseña en una ocasión posterior para autenticarse. Aunque los perfiles no son exactamente iguales, hay bastante similitud y de hecho el cálculo del porcentaje de coincidencias nos arroja un 89%.

En comparación, la figura 2 muestra otra vez el perfil original en rojo, pero ahora acompañado de un perfil en azul generado cuando una persona distinta introdujo la misma contraseña. Podemos apreciar que aunque el patrón general es similar, el ritmo de los tiempos es distinto. El porcentaje de coincidencias es de 59%, lo cual es suficientemente bajo como para no autenticar a la persona en cuestión.

El factor de tolerancia usado podría variar dependiendo de qué tan estrictos queramos ser, pero yo considero que entre un 65 y 75% sería adecuado en la mayoría de los casos.

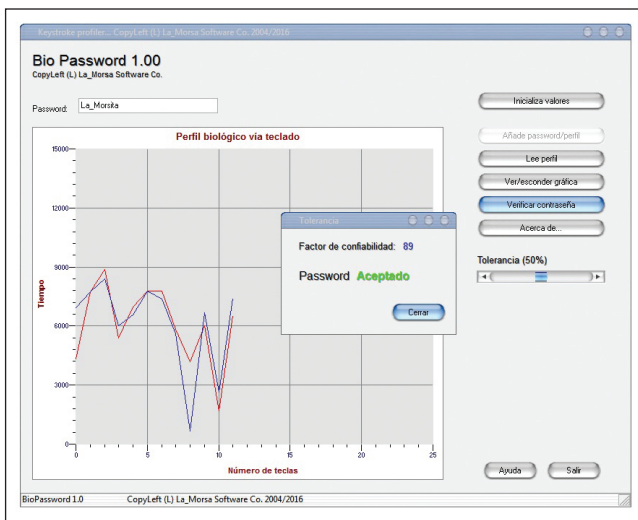


Figura 1. Comparación de perfil de misma persona.

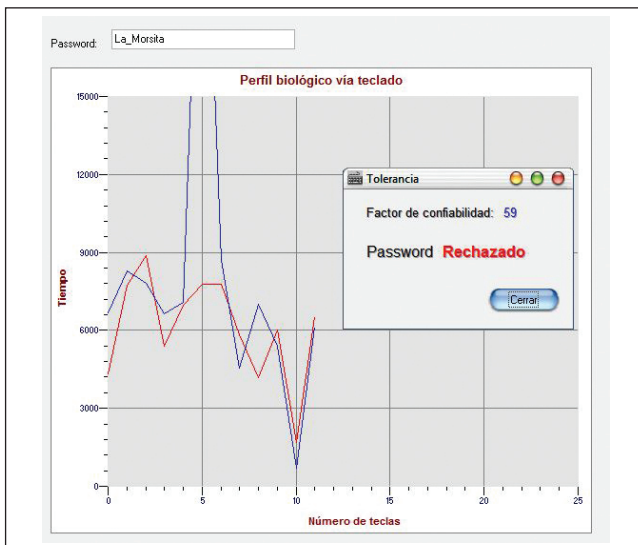


Figura 2. Comparación de perfil de personas distintas.

Hay que indicar que para probar que este esquema de perfiles digitales biométricos, basados en el retardo entre teclas, pueda

validar a usuarios legales de impostores, se necesitan pruebas estadísticas considerando todo género de usuarios. Sin embargo, podemos concluir que el esquema del análisis de las pendientes, considerando solamente si son positivas o negativas da un buen acercamiento a la validez de la idea. Desde luego que es posible generar más parámetros de comparación, así como una incertidumbre de tolerancia, en donde una pendiente, por ejemplo, comparada contra la del perfil, sea igual en 410%, por ejemplo. De esta manera, se estaría incorporando más precisión a la comparación, evitando así que un impostor pudiese, incluso conociendo el método de verificación usado, duplicarlo.

Es evidente que si un usuario válido no se encuentra en las condiciones adecuadas para usarlo (está enfermo, drogado, etc.), el sistema podría detectarlo como un usuario inválido y así evitar que se introdujera al mismo. Más que un inconveniente, esto puede ser útil para restringir acceso a sistemas con información muy sensible.

Cabe señalar que las pruebas se hicieron con un teclado estándar de computadora, pero es obvio que el perfil de un usuario y su contraseña puede ser muy diferente si se usa un teclado de laptop o computadora portátil. A la fecha no tenemos ninguna respuesta a esta problemática. Esto quiere decir que el perfil de un usuario a través de la dinámica del teclado es susceptible del tipo de teclado que se está usando. Quizás en este caso la única manera de soslayar esta dificultad sea subir el nivel de tolerancia sobre el perfil alimentado contra el que pueda dar el usuario al alimentar la contraseña.

Conclusión

Un sistema basado en contraseñas y perfiles dinámicos del teclado puede incrementar la seguridad de los sistemas de cómputo. Sin embargo, entendemos que se necesita muchas más experiencias prácticas para validar los beneficios de esta idea.

A quien le interese esta idea y el programa de demostración, puede pedírmelo a mi correo electrónico: morsa@la-morsa.com, el cual recibirá de forma gratuita por el mismo medio (incluyendo el código fuente en Delphi). ☺

Referencias

- [1] F. Bergadano, D. Gunetti & C. Picardi. "User authentication through keystroke dynamics". *ACM Transactions on Information and System Security* Vol. 5 Issue 4. ACM, 2002.
- [2] D. Chudá, M. Durfina. "Multifactor authentication based on keystroke dynamics". *CompSysTech '09 Proceedings of the International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing*. ACM, 2009.
- [3] R. Joyce, G. Gupta. "Identity authentication based on keystroke latencies". *Communications of the ACM*, Vol. 33 Issue 2. ACM, 1990.

SG[®]talento

Donde se encuentran los profesionistas que crean software grandioso



BENEFICIOS DE TENER TU PERFIL EN SG TALENTO:

- Muestra tu experiencia y habilidades en tu perfil público
- Contribuye a generar información estadística sobre el talento de TI en nuestra región.
- Si estas buscando oportunidades laborales, podrás contactar con reclutadores y aplicar a las vacantes disponibles.
- Además podrás participar en promociones especiales organizadas por SG: descuentos, rifas, concursos, entre otros.

www.sgtalento.com

En Defensa del Software

Por Mike Gregoire

Mike Gregoire es CEO de CA Technologies.

● Recientemente me encontré con dos artículos distintos sobre la industria del software. Uno decía que el mercado del “Internet de las cosas” valdrá 7.1 trillones de dólares para el año 2020. El otro era un informe indicando que hackers habían accedido a datos personales de 15 millones de usuarios de una importante empresa de telecomunicaciones.

A raíz del escándalo de las emisiones de diésel de Volkswagen, en el cuál se utilizaron dispositivos de desactivación y software para perpetuar un fraude, algunos se centran en los aspectos negativos asociados con el software. Así que los resultados de emisiones falsas se han unido al acoso por internet y al robo de identidades electrónicas en la lista cada vez mayor de plagas digitales que corroen la sociedad moderna. Sin embargo, eso es solo una pequeña parte de la historia. La historia más importante es el notable grado en el que el software se ha convertido en el centro de nuestras vidas, y una parte importante de la economía global.

En la economía de las aplicaciones de la actualidad, el software ejecuta todo, desde empresas gigantes hasta tu smartphone. La mayoría de nosotros ahora compra todo tipo de cosas en línea, desde boletos de avión hasta el supermercado.

En el ámbito empresarial, el software permite a las pequeñas empresas escalar rápidamente; acelera procesos y elimina barreras transaccionales; habilita la automatización; genera datos objetivos que pueden ser extraídos para impulsar innovación y mejorar la funcionalidad. Se está abriendo la puerta a modelos de negocio completamente nuevos.

Empresas como Google, Facebook, Microsoft, SAP, Uber, Oracle, Airbnb, Amazon, IBM, Salesforce, Twitter, Yahoo, eBay y CA —por nombrar algunos— no se hubieran podido crear sin el software. Estas 14 compañías por sí solas emplean a más de un millón de personas, y sus ingresos suman más que el PIB total de Austria.

El impacto del software en la tecnología automotriz ofrece un caso revelador al respecto. Los nuevos productos digitales en tu automóvil no sólo sirven para ofrecerte buenas direcciones; estos también hacen que tu coche funcione mejor. Hace muchos años los automóviles dejaron de tener carburadores, que son dispositivos mecánicos, y fueron reemplazados por sistemas de inyección de combustible controlados por software. El software puede hacer mejor el trabajo —ajustando mezclas de combustible de aire con alta precisión para adaptarse a la velocidad del automóvil, la carga del motor, condiciones atmosféricas y las preferencias del operador.

Ese alto nivel de precisión y flexibilidad es lo que hizo posible que unas cuantas personas en Volkswagen “hicieran trampa”. Pero también es lo que hace que tantas nuevas tecnologías sean más rentables, eficientes en recursos y poderosas que sus predecesores mecánicos y analógicos.

En el contexto de cualquier proceso imaginable, el software habilita respuestas delicadas y variadas a una compleja gama de entradas, de una manera que los mecánicos hojalateros de antaño nunca podrían haber soñado. En este momento, muchas de las mentes más brillantes de todo el mundo están afanosamente escribiendo software para resolver los principales problemas que enfrenta la humanidad.

Es mucho más probable que los Thomas Edisons y Alexander Graham Bells de nuestra era se encuentren en un hackathon y no en un taller mecánico. Esto no debe interpretarse en el sentido de que la ingeniería mecánica es un arte que esté muriendo. Lo que significa es que los ingenieros ya están acostumbrados a emplear software para brindar inteligencia o valor agregado a los dispositivos mecánicos.

El software es ahora el rey. Está cambiando la forma de nuestras ciudades, la forma en que nos comunicamos, cómo viajamos. Impulsa el crecimiento económico, mejora la forma en la que el gobierno trabaja y da resultados como, una mejor atención médica. Se puede encontrar en todas partes en grandes cantidades. Los sistemas de control del avión de combate F-22 Raptor contienen más de 8 millones de líneas de código. El Chevrolet Volt, utiliza 10 millones de líneas de código. Mientras que la historia ha demostrado que es posible construir aviones de combate y automóviles eléctricos sin necesidad de utilizar ningún código en absoluto, es bastante claro que ninguno de estos asombrosos vehículos podrían lograr la mitad de lo que hacen sin el beneficio de un software innovador.

Cada que surge una nueva tecnología, hay quienes tratan de aplicarla para fines destructivos o fraudulentos. Pero no por eso culpamos a Gutenberg de que nos llegue folletería promocional por correo, o a los hermanos Wright por las políticas de equipaje de las aerolíneas. Por supuesto que todos tenemos buenas razones para sentirnos engañados por las revelaciones que surgieron del caso de Volkswagen, desencadenadas por el juicio erróneo de un pequeño grupo de empleados. Sin embargo, eso no cambia el hecho de que el software continuará existiendo para hacer del mundo un lugar mejor. 🌍

El Software Orientado al Consumidor y Cómo Cambiará el Panorama para Siempre

Este artículo forma parte del reporte "Trend Watch 2016" publicado por National Instruments y disponible en <http://www.ni.com/trend-watch>.

● **Hace tiempo**, cuando los protectores de camisa para bolígrafos eran funcionales y modernos (un ingeniero puede soñar, ¿no?), el software de ingeniería y el software comercial eran radicalmente distintos—no solo por la gente que los utilizaba sino también por la forma en que se usaban.

Sin embargo, ¿qué sucede cuando una fuerza con gran dinámica (software comercial) se encuentra con un objeto inmóvil (software de ingeniería)? En el mercado actual, que demanda lo mejor de dos mundos, los proveedores de software deben atender ese llamado.

LA EVOLUCIÓN DEL SOFTWARE DE CONSUMIDOR

La llegada del iPhone en 2007 vino a revolucionar la experiencia de uso y expectativas del software comercial gracias a su interfaz simple y fluida. Adicionalmente, su experiencia de compra y comunidad conectada expuso un nuevo mundo de extensibilidad al presentar una interfaz para otra funcionalidad que no ofrecía el proveedor primario. Su huella en la sociedad ha resultado tan indeleble que incluso nuestras generaciones más jóvenes se acercan a los monitores de escritorio y esperan interactuar con ellos de forma táctil.

"Los mismos conceptos que impulsan el movimiento Maker han rejuvenecido la ingeniería. En lugar de trabajar con protocolos complejos que solo pueden entender los ingenieros, el software actual debe proporcionar una interfaz accesible y comprensible para los que no son ingenieros a nivel conceptual".

—Bob O'Donnell, jefe analista de
TECHanalysis Research

LA EVOLUCIÓN DE LAS DEMANDAS EN EL SOFTWARE DE INGENIERÍA

Los usuarios de software de ingeniería del pasado típicamente se graduaban de la universidad con conocimiento de un solo lenguaje de programación. No era raro que se metieran a bajo nivel para manejar manualmente la memoria o los hilos (*threads*) de ejecución. El software era difícil e inabordable para quienes se atrevían a entrar sin el entrenamiento adecuado.

Pero de repente las cosas empezaron a cambiar. En primer lugar, los ingenieros requirieron conocer una amplia variedad de lenguajes de programación para poder enfrentar los retos laborales. Al igual que un guerrero entrenado que cambia de espada a hacha y arco, el ingeniero actual puede entrar y salir de Python, C#, HTML, LabVIEW y Swift. Obligando a que estos lenguajes y herramientas sean accesibles sin necesidad de conocimiento previo o entrenamiento formal.

En segundo lugar, el costo de acceder y adquirir datos ha disminuido con rapidez, mientras que aumenta la necesidad de datos. Y conforme la tecnología está más conectada, ha disminuido el costo de los procesadores. Según DataBeans, el precio de un procesador disminuyó en más de un 30 por ciento de 2011 a 2015. Esto ha acelerado la necesidad de software muy accesible al introducir a más programadores "no tradicionales" en los mundos de la robótica, automatización del hogar e incluso análisis y adquisición de datos generales. Igualmente, las tendencias culturales como el movimiento Maker y la aparición de empresas nuevas de productos comerciales que son adquiridas por cantidades absurdas de dinero son otro ejemplo de este cambio.

UNA CONVERGENCIA INEVITABLE

Para los ingenieros, que se definen por el orgullo de conquistar retos complejos, esta confluencia de usabilidad y sofisticación técnica no podía llegar en mejor momento. Al no tener que distraerse en detalles de herramientas y técnicas de programación, ahora pueden volver a centrarse en grandes retos de ingeniería. En este nuevo futuro, los ingenieros pueden encontrar modos mejores y más rápidos de ir del punto A al B en lugar de perder tiempo realizando mejores mapas.

Igualmente, esta convergencia significa que los ingenieros pueden avanzar hacia un futuro en el que no son los únicos dueños de la innovación. Con software fácil de utilizar, el resto del mundo está poniéndose al día. Además, al reconocer la prevalencia de la simplicidad y la belleza en software de todo tipo, cada vez más gente inteligente desempeñará papeles significativos en la resolución de problemas importantes.

No se puede negar que nuestro mundo cada vez es más complicado y que nuestros retos cada vez son más difíciles de resolver. El que continuamente se nos solicite hacer más con menos dinero, menos gente y menos tiempo, solo incrementa la tensión. Cuanto antes lleguemos a la convergencia del software de ingeniería y del software comercial, antes nos daremos cuenta de que lo "simple" es el mejor modo de resolver lo "complejo". ☺

Entendiendo MapReduce

Por Pedro Galván

● **MapReduce es un modelo** de programación para resolver problemas por medio del procesamiento paralelo de grandes cantidades de datos en un clúster de cómputo.

En esencia, el algoritmo MapReduce está compuesto de un procedimiento de mapeo (Map) y otro de reducción (Reduce). El mapeo se enfoca en filtrar y organizar los datos (por ejemplo tomar datos de estudiantes y organizarlos en grupos de acuerdo a su nombre de pila) mientras que la reducción se dedica a combinar y sumarizar datos (por ejemplo contar el número de estudiantes que hay en cada grupo identificado).

Si MapReduce es un modelo, ¿por qué es común referirse a éste como una tecnología o implementación? Lo que sucede es que para ejecutar tareas de tipo MapReduce se requiere de un sistema de software que orqueste el procesamiento paralelo, gestionando la comunicación y transferencia de datos entre los distintos nodos de procesamiento. Así que MapReduce es un modelo de programación acompañado de una implementación. MapReduce fue creado (tanto el modelo como la implementación original) por Google y presentado al público en 2004.

La principal ventaja de MapReduce es que facilita el procesamiento paralelo masivo. Definir correctamente las tareas de map() y reduce() es algo no necesariamente trivial, pero una vez que se logra esto se puede repartir la carga de trabajo hacia cientos o miles de nodos de procesamiento de forma sencilla y transparente.

EL ALGORITMO

Un programa MapReduce se ejecuta en tres etapas: map, shuffle y reduce.

Map: En esta etapa se toman los datos a partir de una entrada y se procesan línea por línea por la función de mapeo. Ésta procesa los datos y los organiza en grupos. Los datos que genera como salida están en forma de tuplas de pares llave-valor.

Shuffle: Consiste en distribuir los datos a procesarse entre los distintos nodos. Dicha asignación se hace basándose en la llave correspondiente, de manera que todos los datos pertenecientes a cierta llave se asignen al mismo nodo de procesamiento.

Reduce: Es cuando se procesan los datos en paralelo a través de los distintos nodos, cada nodo se dedica a combinar/reducir los datos que le fueron asignados para crear un nuevo conjunto de datos sumariado.

Como ya comentamos, el framework MapReduce opera en base a pares <llave, valor>. Es decir, las entradas de datos son vistas como conjuntos de pares <llave, valor> y el resultado de procesarlos es un nuevo conjunto de pares <llave, valor>, posiblemente con otro tipo de datos.

La tabla 1 refleja la estructura de los pares de datos al entrar y salir de cada etapa.

	Entrada	Salida
Map	<k1, v1>	list (<k2, v2>)
Reduce	<k2, list(v2)>	list (<k3, v3>)

UN EJEMPLO

Veamos un ejemplo sencillo. Asumiendo que tenemos cinco archivos, y cada archivo contiene listas de datos organizados en dos columnas (una llave y un valor) que representan una ciudad y una temperatura registrada en dicha ciudad. El contenido de un archivo podría ser el siguiente:

```
México,20
Lima,25
Bogotá,22
Buenos Aires,32
México,4
Buenos Aires,33
Bogotá,18
```

Lo que queremos encontrar es la máxima temperatura para cada ciudad a través

de todos los archivos de datos (es posible que una ciudad tenga varias ocurrencias de datos en cada archivo). Usando MapReduce, podemos dividir esto en 5 tareas de mapeo, donde cada mapeador trabaja en un archivo de entrada.

El resultado de procesar el contenido del archivo previamente indicado sería el siguiente:

```
<México,20> <Lima,25> <Bogotá,22> <Buenos Aires,33>
```

Imaginemos que el resultado de procesar los otros 4 archivos es el siguiente:

```
<México,18> <Lima,27> <Bogotá,32> <Buenos Aires,37>
<México,32> <Lima,20> <Bogotá,33> <Buenos Aires,38>
<México,22> <Lima,19> <Bogotá,20> <Buenos Aires,31>
<México,31> <Lima,22> <Bogotá,19> <Buenos Aires,30>
```

Todos estos resultados se podrían alimentar de nueva cuenta a la tarea de reducción, que combinaría los distintos datos y generaría un valor único para cada ciudad. El resultado sería:

```
<México,32> <Lima,27> <Bogotá,33> <Buenos Aires,38>
```

MapReduce opera de forma análoga a como se realizan los censos de población: se despachan agentes a las calles, estos visitan las casas haciendo el conteo en cada una, y regresan los resultados que son concentrados para obtener la población del área estudiada. Es mucho más rápido tener múltiples agentes trabajando en paralelo (por ejemplo uno en cada calle) que tener un solo agente que recorra todas las casas de forma serial. ☹

Referencias

[1] IBM - What is Map Reduce? <http://swqu.ru/ra>

[2] Tutorialpoint: Hadoop - MapReduce. <http://swqu.ru/rb>

UX MASTERCLASS 2016 POTENCIANDO LA MADUREZ DE UX

CIUDAD DE MEXICO, ABRIL 27



Asiste a este gran evento internacional que reúne a expertos y líderes de la industria UX para compartir sus conocimientos



15 conferencias relacionadas con madurez de UX en las organizaciones



Expertos locales e internacionales



Cocktail y network con colegas



Jorge Guevara
Vicepresidente de Asuntos Corporativos y Comunicación
@ American Express

Robert Shummacher
Vicepresidente Ejecutivo
@ GFK User Centric

Frederic Gaillard
Director General
@ Axance

Luis Carlos Aceves
Director General
@ Usaria

Annemarie Lesage
Directora de Investigación y Diseño UX
@ YU Centrik

Helga Stegmann
Directora General
@ Mantaray

Simon Herd
Director de Investigación
@ Sutherland Labs

Inscríbete hoy e interactúa con los principales exponentes de UX a nivel mundial

\$3,500 MXN
PERSONAL

\$3,000 MXN*
GRUPAL

Comida e IVA incluidos. *4 o más personas, precio por persona.

ESCUCHA LA VISIÓN UX DE CONFERENCISTAS DE 10 PAÍSES DISTINTOS

**ADQUIERE HOY
TUS BOLETOS**

www.uxmasterclass.com

CIUDAD DE MÉXICO ☒ ABRIL, 27 DE 2016



ORGANIZADO POR:



+ UX alliance

1

HOLOLENS DEVELOPMENT EDITION



Los desarrolladores interesados en construir aplicaciones para HoloLens ya pueden adquirir el kit para desarrolladores. A diferencia del Oculus Rift, que requiere estar conectado a una computadora, HoloLens es un dispositivo autocontenido y soberano. Sus lentes cuentan con un mecanismo de proyección de imágenes holográficas, de manera que cuando el usuario ve a través de estos percibe los hologramas como una capa de realidad aumentada. HoloLens cuenta con diversos sensores, 4 micrófonos y 6 cámaras, incluyendo un sensor de profundidad, a través de los cuales percibe el mundo alrededor del usuario. HoloLens cuenta con dos unidades de procesamiento: un procesador Intel de 32 bits de propósito general, y un procesador holográfico diseñado por Microsoft con este propósito específico. Ahora que ya te emocionamos, podemos darte los detalles no tan agradables. El primero es que este kit tiene un precio de 3,000 dólares, así que no es para cualquiera. Adicionalmente, para desarrollar se requiere una PC con Windows 10, Visual Studio 2015 y Unity 5.4.

2

AXIS VIDUIS

El Vidius es un dron creado por la empresa Axis Drones pretende ser el dron más pequeño del mundo con vista en primera persona (y hasta donde sabemos, lo logra). Este cuadricóptero mide tan solo 4.3 cm x 4.3 cm x 2.5 cm y es capaz de volar hasta a 30 metros de distancia mientras graba y transmite video por streaming. Cuenta con una cámara de 1 megapixel que captura video con calidad 480p, lo cual no es nada grandioso pero puede ser suficiente para cosas sencillas. Vidius se conecta por WiFi a tu smartphone Android o Apple para que puedas controlarlo, y cuenta con una batería de 150 mAH que le da 5 minutos de vuelo.



3

RASPBERRY PI 3 MODEL B

La comunidad maker fue gratamente sorprendida con el lanzamiento del Raspberry Pi 3 Model B, apenas un año después de haber lanzado el Pi 2. La principal mejora del Raspberry Pi 3 es que usa un procesador de 64 bits ARM Cortex A53 con una velocidad de reloj de 1.2 GHz, lo cual lo hace hacer cerca de 50% más rápido que el Pi 2. Adicionalmente, el Pi 3 ahora tiene integrada conectividad WiFi 802.11n y Bluetooth 4.0, por lo que podrás conectar teclados y mouses inalámbricos sin necesidad de habilitar accesorios extra. La memoria RAM se mantiene en 1 GB y las capacidades de video también se mantiene en VideoCore IV, aunque ahora operando a 400 MHz. Lo mejor de todo esto es que el Pi 3 tiene el mismo precio que su predecesor, 35 dólares. Así que si tenías pendiente comprar un Pi, este es el momento.

4 ROBOHON Visto en @geek_mx

¿Siempre quisiste tener tu propio y fiel androide? Pues tus plegarias han sido escuchadas. RoBoHon es un pequeño robot de 7 pulgadas desarrollado por Sharp el cual hace llamadas, recibe e-mails, descarga aplicaciones e, incluso, baila al ritmo de tu música favorita. Por si esto fuera poco, dicho dispositivo también cuenta con un proyector en el que puedes ver tus fotos y películas. Aparentemente, RoBoHon no es simplemente un smartphone con apariencia de robot; sino que realmente se encuentra diseñado para funcionar como un pequeño mayordomo computarizado.



5 HP ZBOOK STUDIO

HP recientemente lanzó la 3er generación de su línea de estaciones de trabajo móviles ZBook, y debemos confesar que estamos perdidamente enamorados del nuevo modelo ZBook Studio. Parece ser que reúne todo lo que los usuarios avanzados hemos deseado desde hace años: una estación de trabajo portátil que ofrezca el más alto desempeño de procesamiento, pero que al mismo tiempo sea elegante y verdaderamente portátil. Por fuera, la ZBook Studio es una computadora elegante y delgada (18 mm), con una pantalla de 15.6" con dos opciones de resolución: 1080p o 4K. Pero la discreción queda solamente por fuera, porque por dentro la ZBook es un monstruo: puede tener procesador Intel Core i7 o Xeon E3, dos unidades de disco para ofrecer hasta 2 TB de almacenamiento, hasta 32 GB de RAM DDR4 a 2133 Mhz, tarjeta de video NVIDIA Quadro M1000M con 4 GB de memoria dedicada, y dos puertos Thunderbolt. Adicionalmente, la ZBook Studio fue diseñada para satisfacer especificaciones militares STD 810G de resistencia a golpes y condiciones extremas.

Humor



Original: <http://geek-and-poke.com/geekandpoke/2014/11/8/frameworks>

¿Cómo Llegamos Aquí?

● Preparar el número 50 de **Software Guru**, inevitablemente nos ha llevado a la nostalgia. Y esa nostalgia nos llevó a preguntarnos. ¿Cómo han cambiado las cosas en nuestra industria y profesión durante el tiempo en que se han publicado estos 50 números? (poco más de 11 años).

Presentamos a continuación un análisis cronológico relacionando los temas principales de SG durante sus primeros 3 años, con lo que vemos hoy en día.

0. Octubre 2004

El número de estreno de SG se enfocó en la industria de software en México y el recién horneado programa ProSoft. Hanna Oktaba participó como columnista desde entonces. Desde este número ya hablábamos de Usabilidad del software.



1. Enero 2015

La fiebre de los modelos formales de madurez de procesos estaba a todo lo que daba, así que desarrollamos una guía para explicar las características distintivas de los principales modelos de procesos. Fue el primer número con Luis Cuellar como columnista.



2. Marzo 2015

El concepto de “fábricas” o Centros de Desarrollo de Software se hace popular, y estudiamos este modelo.

Entrevistamos a Blanca Treviño, CEO de Softtek, quien ya era reconocida pero no con el alcance que tiene hoy.



3. Mayo 2015

El software libre todavía luchaba por ser reconocido como una postura válida. Hoy es el común denominador e incluso Microsoft está haciendo .Net software libre. En este número, Luis Vinicio León, Raúl Trejo y Luis Daniel Soto iniciaron su participación como columnistas.



4. Julio 2005

Los sistemas BPM (Business Process Management) se basan en el concepto de modelar visualmente procesos de negocio y utilizar motores de workflow para automatizarlos. Actualmente todavía existen y mantienen su nicho.



5. Septiembre 2005

Antes de los smartphones, el iPhone, las app stores y todo lo demás, ya había dispositivos de cómputo móvil (¿se acuerdan de las Palm Pilot?), y SG estuvo ahí para guiarnos a desarrollar aplicaciones (todavía no se llamaban “apps”).



6. Noviembre 2005

¿Qué es la mejora de procesos de software y con qué objetivos deberíamos establecer una iniciativa de este tipo? En este número apareció nuestra primera encuesta de salarios, que se convertiría en referencia de la industria.



7. Enero 2006

Hace 10 años no había big data; había OLAP (Online Analytical Processing) y su paradigma de cubos de datos. Estrenamos la columna de UML a cargo de nuestros amigos de Milestone Consulting (hoy Abitzar) que continúan apoyando a SG.



8. Marzo 2006

Criptografía, modelado de amenazas, transferencias de datos seguras, y utilización de biometría.

Entrevista con el mismísimo Jon “maddog” Hall cuando vino a ExpoLinux que se hizo junto con ExpoComm (ya llovió).





9. Mayo 2006

Érase una vez que los métodos ágiles eran vistos como algo revolucionario y utópico. El método más conocido (aunque no por ello más usado era eXtreme Programming), e iniciaba algo llamado Scrum que parecía tener futuro.



10. Julio 2006

Las aplicaciones empresariales (ERP) no son lo más glamoroso de nuestra profesión, pero no podemos ignorarlas. Visual Studio pasa de ser un IDE, a una suite de desarrollo colaborativo. Hablamos por primera vez de arquitectura orientada a servicios (SOA).



11. Septiembre 2006

Hace tiempo, cuando todavía desconocíamos la existencia de los hipsters y nadie se atrevía a usar corbatas de moño, hablamos sobre usabilidad y user experience. También, la guerra entre "javeros" y "dotneteros" era intensa, y publicamos un comparativo entre estas plataformas.



12. Noviembre 2006

El testing de software es parte esencial del desarrollo de software. Sin embargo, no hace tanto que se le comenzó a profesionalizar en nuestra región. Esta fue una de las primeras ocasiones en las que lo abordamos formalmente.



13. Enero 2007

En el mundo pre-iPhone, eran muy pocos los que tenían la suerte de poder desarrollar videojuegos, debido al costo de las herramientas y las dificultades de distribución. Afortunadamente, hoy herramientas como Unity lo hacen más sencillo, y las app stores resuelven distribución.



14. Marzo 2007

Más allá de la evolución (o falta de) que han tenido los lenguajes de programación en la última década, el toolchain de herramientas de control de configuración y ensamble de aplicaciones ha avanzado enormemente. Fue en este número cuando le dedicamos atención a esto.



15. Mayo 2007

Durante la primera mitad de la década de los 2000, eran pocas las empresas que hacían desarrollo offshore (nears-hore) desde Latinoamérica, pero en la segunda mitad esto se incrementó drásticamente, y hoy es de lo más común.



16. Julio 2007

La primer generación de la web fue unidireccional, había una clara división entre productores y consumidores de contenido y la comunicación solo iba en un sentido. Luego llegaron las wikis, los sitios de upvoting, las redes sociales, y todo cambió.



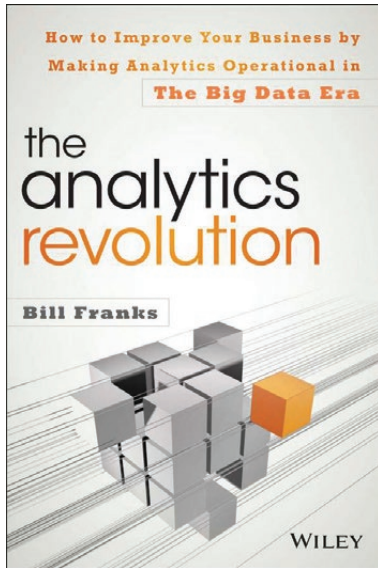
17. Septiembre 2007

Todavía no se utilizaba el término "Internet de las cosas", pero ya se vislumbraba la oportunidad usar software y sensores para dotar de cierta inteligencia e interactividad a dispositivos físicos. El protocolo de comunicación Zigbee tomaba fuerza y publicamos al respecto.



18. Noviembre 2007

Publicamos sobre una nueva tecnología llamada Ruby on Rails que ganaba popularidad. No imaginábamos que este inocente framework desencadenaría una ola de lenguajes dinámicos y frameworks automáticos, así como el surgimiento del desarrollador hipster. ☺



THE ANALYTICS REVOLUTION: HOW TO IMPROVE YOUR BUSINESS BY MAKING ANALYTICS OPERATIONAL IN THE BIG DATA ERA

1

Bill Franks. Wiley, 2014.

The Analytics Revolution está dirigido principalmente a gerentes y directivos interesados en establecer una práctica exitosa de analítica de datos en su organización.

A lo largo del libro se abordan distintos aspectos, tales como:

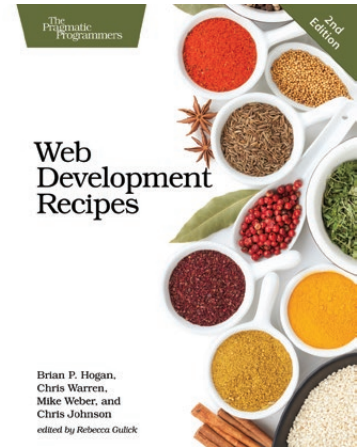
- Panorama de qué es la analítica operacional y las tendencias que nos han llevado hasta aquí.
- Cómo estructurar un equipo de analytics que cubra roles de análisis, desarrollo e infraestructura.
- Cómo cambiar la cultura corporativa para habilitar el descubrimiento de información basado en analítica y la reducción de tiempos para implementar mejoras basadas en la información descubierta.
- Guías sobre cómo justificar, implementar y gobernar una práctica de analítica.

El autor, Bill Franks, es director de analytics en la empresa Teradata, así que tiene bastante experiencia en el tema, así como exposición a casos reales en organizaciones complejas.

WEB DEVELOPMENT RECIPES, 2ND EDITION.

Brian Hogan, Chris Warren, Mike Weber, Chris Johnson. Pragmatic Bookshelf, 2015.

2



El desarrollo web está en constante evolución, y los desarrolladores requieren mantenerse al tanto de las más recientes técnicas y tecnologías. El propósito de Web Development Recipes es justamente ayudar a los desarrolladores web a mantener actualizado su repertorio de herramientas.

La primera edición de Web Development Recipes se publicó a principios de 2012, y esta 2da edición se publicó en julio de 2015, así que está bastante fresca y aborda herramientas como Vagrant, Grunt, Angular y React.

El libro está organizado de acuerdo a tipos de recetas. A continuación listamos las secciones y principales recetas:

- Estilizado visual: estilizado de botones, animaciones con CSS3, íconos tipográficos.
- Interfaz de usuario: handlebars templating, paginación sin fin, knockout.js, React, Angular.
- Datos: agregar mapas en línea, datos cross-site con JSONP, widgets.
- Móvil: menús drop-down responsivos, drag and drop en móviles.
- Workflow: diseño responsivo con Skeleton, SaSS, Git.
- Testing: Selenium, Nightwatch, Jasmine.
- Ambiente de ejecución: despliegue con Grunt, configuración de máquinas virtuales con Puppet.