

REVISTA DEL DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN DE LA UNIVERSIDAD DE CHILE

Bits DE CIENCIA

PRIMER SEMESTRE 2016

INDUSTRIA DEL SOFTWARE Y TIC EN CHILE

¿EXISTE UNA INDUSTRIA TI EN CHILE? | José Miguel Piquer

RADIOGRAFÍA A LA INDUSTRIA DEL SOFTWARE EN CHILE Y SUS DESAFÍOS | María Eugenia Riquelme



fcfm

Ciencias de la Computación
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

13



COMITÉ EDITORIAL

Claudio Gutiérrez, profesor
Alejandro Hevia, profesor
Gonzalo Navarro, profesor
Sergio Ochoa, profesor

EDITOR GENERAL

Pablo Barceló, profesor

EDITORIA PERIODÍSTICA

Ana Gabriela Martínez

PERIODISTA

Marta Apablaza

DISEÑO

Puracomunicación

FOTOGRAFÍAS E IMÁGENES

Comunicaciones DCC UChile
Comunicaciones DCC PUC
Puracomunicación

Revista BITS de Ciencia del Departamento de Ciencias de la Computación de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile se encuentra bajo Licencia Creative Commons Atribución-NoComercial-Compartir-Igual 3.0 Chile. Basada en una obra en www.dcc.uchile.cl



Revista Bits de Ciencia N°13

ISSN 0718-8005 (versión impresa)

www.dcc.uchile.cl/revista

ISSN 0717-8013 (versión en línea)



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE CHILE

Departamento de Ciencias**de la Computación**

Avda. Beauchef 851, 3º piso,
edificio norte
Santiago, Chile
837-0459 Santiago
www.dcc.uchile.cl
Fono 56-2-29780652
Fax 56-2-26895531
revista@dcc.uchile.cl

CONTENIDOS

01 EDITORIAL

| Pablo Barceló

ESPECIAL ANIVERSARIO 40 AÑOS DCC



02 EL DCC CONMEMORÓ 40 AÑOS DE VIDA

INVESTIGACIÓN DESTACADA



08 EL DILUVIO DE DATOS Y LA REPETITIVIDAD: NUESTRO MEJOR ENEMIGO

| Gonzalo Navarro

COMPUTACIÓN Y SOCIEDAD

16 EMPRESA NACIONAL DE COMPUTACIÓN E INFORMÁTICA (ECOM) 1971-1973

| Juan Álvarez

LA INDUSTRIA DEL SOFTWARE Y TIC EN CHILE



28 ¿EXISTE UNA INDUSTRIA TI EN CHILE?

| José Miguel Piquer

31 RADIOGRAFÍA A LA INDUSTRIA DEL SOFTWARE EN CHILE Y SUS DESAFÍOS

| María Eugenia Riquelme

34 LA INDUSTRIA DEL SOFTWARE EN CHILE: DESAFÍOS Y OPORTUNIDADES PARA GENERAR UN IMPACTO PAÍS

| María Cecilia Bastarrica

| Maira Marques

| Sergio Ochoa

| Daniel Perovich

| Alcides Quispe

| Romain Robbes

| Jocelyn Simmonds

40 DATA SCIENCE APLICADA AL DESARROLLO DE PRODUCTOS

| Pablo García

44 RED CHILENA DEL MEJORAMIENTO DEL PROCESO DE SOFTWARE, SPIN-CHILE A.G.

| Erika Madariaga



46 POSTÍTULOS EN TECNOLOGÍAS DE LA INFORMACIÓN

| José Pino

| María Cecilia Bastarrica

| Yadran Eterovic

| Jaime Navón

| Raúl Monge

CONVERSACIONES

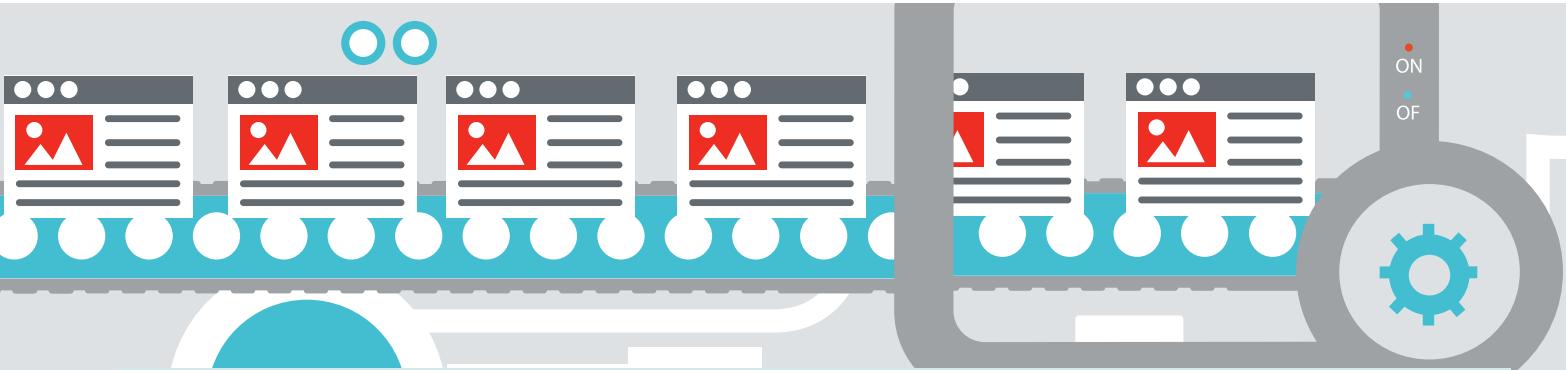
50 ENTREVISTA A ROBERTO OPAZO

| Por Alejandro Hevia

SURVEYS

56 PROPUESTAS DE BUENAS PRÁCTICAS DE DESARROLLO DE SOFTWARE PARA PEQUEÑAS EMPRESAS DE SOFTWARE

| Alcides Quispe



Esta vez tenemos una gran razón para celebrar: nuestro Departamento - el Departamento de Ciencias de la Computación de la Universidad de Chile - ha cumplido 40 años. Lo que comenzó en 1975 como la apuesta liderada por un grupo de jóvenes para instalar a la Computación como una nueva área de la Ciencia y la Ingeniería, ha dado paso hoy en día a un Departamento de punta a nivel latinoamericano en docencia, transferencia e investigación. Orgullosos como estamos de esta historia, comenzamos esta nueva edición de la "Bits de Ciencia" con textos y fotos extraídas de la celebración de estos 40 años, la que se realizó en diciembre de 2015 con presencia de autoridades, académicos, funcionarios, y exalumnos, entre otros.

Por otro lado, el tema principal de la Bits en esta ocasión es la industria del software y TIC en Chile. Hemos congregado a varios expertos de la academia y la industria para que hablen del tema, en algunos casos aportándonos con su visión sobre ésta y en otros relatándonos el trabajo que han realizado para que esta industria crezca y aumente aún más en calidad. Parece ser un tema común a todos estos artículos el hecho que la industria chilena de software y TIC tiene un buen nivel y potencial para desarrollarse, en particular producto de la calidad de sus profesionales. Por otra parte, el tamaño de tal industria es pequeño, y parece necesario desarrollar políticas de gran escala que la impulsen a convertirse en real motor de desarrollo para Chile.

La sección dedicada a la industria nacional del software y TIC incluye lo siguiente:

- Un artículo de José Miguel Piquer que responde a las preguntas de si esta industria realmente existe y por qué no se ha desarrollado plenamente.
- Una radiografía a la industria del software nacional y sus desafíos, realizada por María Eugenia Riquelme de GECHS.
- Un artículo de varios académicos y alumnos de postgrado de nuestro Departamento (proyecto GEMS) analizando los desafíos y oportunidades que esta industria tiene para generar impacto país.
- Un artículo de Pablo García, de Telefónica I+D, contándonos los esfuerzos que están realizando para desarrollar transferencia a través de la combinación TIC/data science.
- Una reseña de Erika Madariaga sobre SPIN-Chile, la red nacional de mejoramiento del proceso de software.
- Por último, una breve descripción de algunos postítulos en el área que se imparten en nuestro país.

Tenemos además nuestras secciones tradicionales:

- En Investigación Destacada, el académico Gonzalo Navarro nos habla sobre su investigación en Big Data y repetitividad.
- En Computación y Sociedad seguimos con la saga del Profesor Juan Álvarez sobre la Historia de la Computación en Chile. En este caso su artículo trata de los primeros años de ECOM, la Empresa Nacional de Computación fundada en 1970.
- En la sección Conversaciones, el académico

Alejandro Hevia entrevista a nuestro exalumno Roberto Opazo sobre cómo fue su experiencia en nuestro Departamento y los orígenes de su capacidad de innovación en diferentes empresas.

- Finalmente, nuestro exalumno de Doctorado Alcides Quispe nos presenta un Survey sobre buenas prácticas de desarrollo de software para pequeñas empresas.

Esperamos que disfruten la Revista. Como siempre quedamos a disposición de sus consultas y comentarios a través de nuestro mail revista@dcc.uchile.cl

PABLO BARCELÓ

Editor General

Revista Bits de Ciencia



EL DCC CONMEMORÓ 40 AÑOS DE VIDA

En una emotiva ceremonia que congregó a egresados, estudiantes, académicos y funcionarios, el Departamento de Ciencias de la Computación (DCC) de la FCFM celebró cuarenta años de historia.





En 1975 un grupo de jóvenes y entusiastas ingenieros creó el Departamento de Ciencias de la Computación (DCC), como una iniciativa que buscaba explorar esta nueva área del conocimiento y abrir oportunidades para el uso de tecnologías digitales en el país. Cuarenta años han transcurrido desde entonces, en los que el DCC ha recorrido un largo camino que fue recordado el 11 de diciembre de 2015 en una ceremonia que contó con la presencia de los fundadores del DCC: Fernando Silva, Alfredo Piquer, José Pino y Patricio Poblete, así como de las autoridades de la FCFM, académicos, funcionarios, alumnos y egresados del Departamento.

Previo a esta ceremonia, y como parte de la celebración, se realizaron diversas actividades. La primera actividad fue un "Puertas Abiertas" que buscó mostrar la carrera de Ingeniería Civil en Computación. Allí los profesores José Miguel Piquer y Nancy Hitschfeld realizaron charlas sobre las tecnologías que revolucionan la humanidad, y sobre cómo la Computación puede ser un campo laboral atractivo para mujeres. Luego el Profesor Ricardo Baeza Yates, académico DCC y Chief Research Scientist Yahoo! Labs, dictó la charla "Big Data en la Web". También se realizaron visitas guiadas a las instalaciones del Departamento donde, entre otras actividades, los asistentes pudieron ver el trabajo que se realiza en el área de robótica y conocer el Museo DCC.

Durante la ceremonia oficial de aniversario, el Director del DCC, Prof. Sergio Ochoa resaltó diferentes hitos del Departamento, como la creación de programas de pre y postgrado, el envío del primer mail, la primera conexión a Internet del país y la creación de NIC Chile. Asimismo, destacó la participación en iniciativas gubernamentales y otras de impacto país, el trabajo de investigación de punta que se ha realizado en las últimas dos décadas, que le han dado visibilidad internacional al DCC y el hecho que en 2015, el ranking QS posicionó al Departamento en el tercer lugar, entre los departamentos de Computación de Latinoamérica, detrás de las universidades de São Paulo y Campinas de Brasil.

"Hoy tenemos un DCC un poco distinto a aquel que creó aquel grupo de ingeniosos ingenieros por allá por 1975 y que hoy le rendimos homenaje. Aquellos que tomamos la posta seguimos compartiendo los mismos valores y poniéndole

el mismo empeño para que nuestra casa sea cada día mejor, más inclusiva, solidaria y preocupada de los problemas de Chile y de su gente", señaló el Prof. Ochoa.

También tomó la palabra el fundador y primer Director del DCC Fernando Silva, quien además de recordar los inicios del Departamento, expresó su orgullo por la evolución del DCC: "Trabajé en la Facultad por 16 años. Tuve el privilegio de trabajar con jóvenes con gran capacidad y verlos impartir sus conocimientos con una generosidad sin límites. Al cumplir 40 años el Departamento, nos llena de orgullo ver que ha llegado a ser uno de los más reconocidos de Latinoamérica".

A continuación, los profesores José Miguel Piquer y Patricio Poblete, pioneros del desarrollo de Internet en Chile, recordaron los años iniciales del DCC. El profesor Piquer afirmó que "de este Departamento ha salido una enorme cantidad de generaciones y de gente que hoy está haciendo la Computación de este país. Comparto con ustedes la nostalgia y emoción de reencontrarnos".

Por su parte, la egresada Marcela Calderón afirmó que ser ingeniera del DCC "es una marca. Más allá del conocimiento y habilidades técnicas destaca el genuino interés por saber más por resolverlo todo y por aprender". Agregó que hoy "el DCC enfrenta grandes desafíos y oportunidades. El primero es el de mantener esta marca. Pero también sería positivo contar con más mujeres en este campo. Atraer más mujeres es una tarea posible y demostrar que ésta es no sólo una carrera para hombres".

Tras estas intervenciones, se realizó un reconocimiento a los fundadores del Departamento: Fernando Silva, Alfredo Piquer, José Pino y Patricio Poblete; a los profesores José Miguel Piquer, Ricardo Baeza Yates, Jorge Olivos y a quien fuera por largo tiempo jefa administrativa, Margarita Serei, todos como artífices del desarrollo del DCC. También se reconoció al primer egresado de cada programa: Ronald Corovic (Ingeniería Civil en Computación), José Benguria (Magíster en Ciencias mención Computación), José Miguel Rodrigo (Magíster en TI), Gonzalo Navarro (Doctorado en Ciencias mención Computación) y Rigoberto Molina (Ingeniería en Ejecución en Procesamiento de la Información).



Cerró la ceremonia el Decano de la FCFM, Prof. Patricio Aceituno, quien reconoció la innovación que hubo de parte de los fundadores del DCC, al vislumbrar la Computación como un área a desarrollar. "Cuando miro 40 años atrás y el re-

nacer del DCC, qué era eso sino innovación. Y para hacerla requiere: dedicación, pasión y mucho esfuerzo. La Universidad de Chile y la Facultad deben reconocer que si este país tiene la capacidad que tiene en Computación e Infor-

mática es por esa pasión, dedicación y esfuerzo que puso un grupo de pioneros que llevó la disciplina al nivel que está en Chile", concluyó la máxima autoridad. ■



1. El Director del DCC, Prof. Sergio Ochoa. | 2. Fernando Silva, fundador y primer Director del DCC. | 3. El Decano de la FCFM, Prof. Patricio Aceituno.
- | 4. Los profesores José Miguel Piquer y Patricio Poblete, recordaron los años iniciales del DCC. | 5. Marcela Calderón, profesora y egresada del DCC.
- | 6. Margarita Serei, Jefa Administrativa del DCC hasta 2011, recibió el reconocimiento "Artífices del Desarrollo del DCC", de manos del Director del DCC, Prof. Sergio Ochoa. | 7. De izquierda a derecha, los profesores José Miguel Piquer y Ricardo Baeza-Yates, recibieron el reconocimiento "Artífices del Desarrollo del DCC", de manos del Director del DCC, Prof. Sergio Ochoa. | 8. Los fundadores del DCC. De izquierda a derecha: Fernando Silva, José Pino, Alfredo Piquer y Patricio Poblete. | 9. Entrega de reconocimiento a los primeros graduados de nuestros programas de estudio. De izquierda a derecha: Ronald Corovic (Ingeniería Civil en Computación), el Director del DCC, Prof. Sergio Ochoa, José Benguria (Magíster en Ciencias mención Computación), José Miguel Rodrigo (Magíster en TI) y Gonzalo Navarro (Doctorado en Ciencias mención Computación).



CELEBRANDO LOS 40 AÑOS EN COMUNIDAD

EN UN EMOTIVO ENCUENTRO GENERACIONAL, EL ANIVERSARIO DEL DCC REUNIÓ A SUS ACADÉMICOS, ALUMNOS, EGRESADOS Y FUNCIONARIOS.



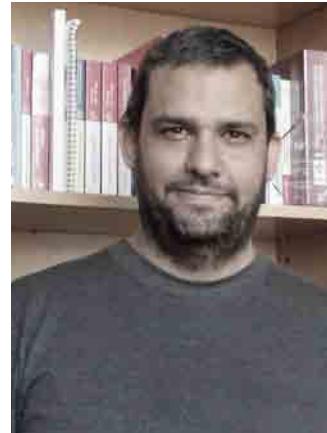
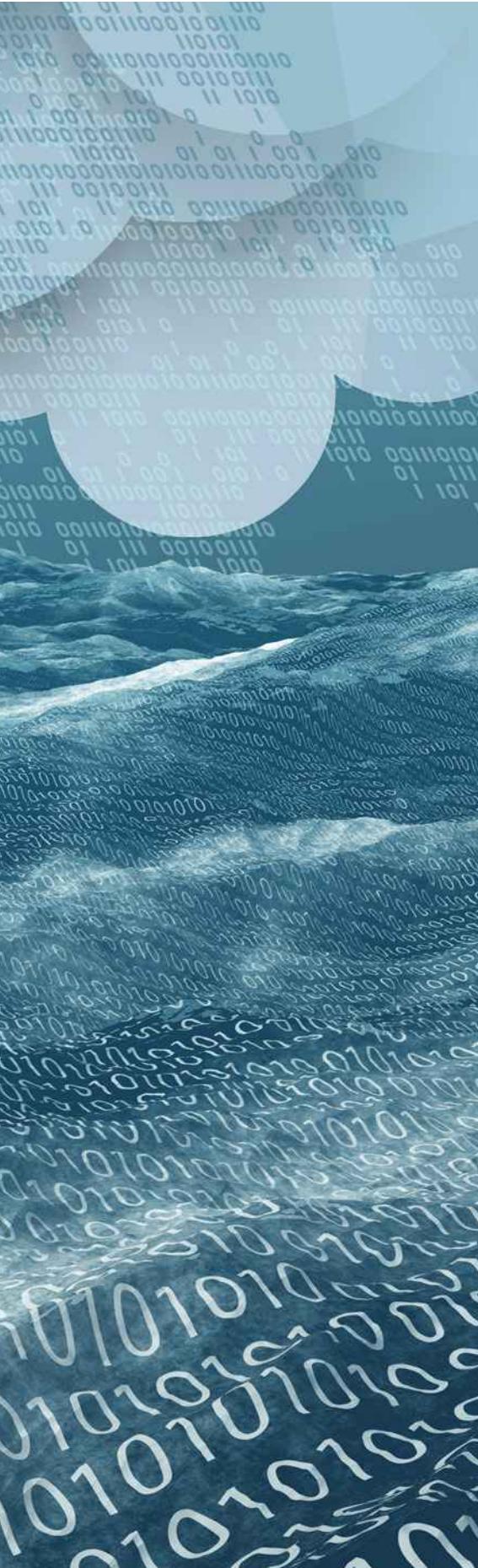




EL DILUVIO DE DATOS Y LA REPETITIVIDAD: NUESTRO MEJOR ENEMIGO*

Uno de los aspectos que mejor caracteriza nuestro desarrollo tecnológico en las últimas décadas es el rápido crecimiento de los datos disponibles digitalmente. En pocos años, hemos pasado de estar preocupados por la cantidad de cómputo que debíamos realizar a preocuparnos más bien por el volumen de datos que debemos procesar, incluso cuando los cálculos son relativamente simples .





GONZALO NAVARRO

Profesor Titular Departamento de Ciencias de la Computación, Universidad de Chile. Doctor en Ciencias mención Computación y Magíster en Ciencias mención Computación, Universidad de Chile; Licenciatura en Informática, Universidad Nacional de La Plata y ESLAI, Argentina. Líneas de investigación: Diseño y Análisis de Algoritmos, Bases de Datos Textuales, Búsqueda por Similitud, Compresión.

gnavarro@dcc.uchile.cl

EL DILUVIO DE DATOS

Tenemos por un lado los datos que podríamos llamar "científicos": los datos astronómicos, donde Chile es una fuente generadora predilecta; datos de clima obtenidos por sensores de todo tipo, que en varios casos deben ser usados en tiempo real para ser útiles; datos geográficos, incorporados a la vida diaria gracias a los satélites e intermediarios como Google Maps; secuencias biológicas, cada vez más baratas de obtener y por ello más abundantes; etc.

Tenemos también datos que podríamos llamar "contenidos": grabaciones de voz, imágenes, fotos, música, videos, libros, revistas, páginas web, emails, blogs, tweets, y un largo etcétera, todos ellos incorporando la necesidad de encontrar el contenido deseado en formatos no siempre amigables para el procesamiento automático.

En una tercera categoría, que podríamos llamar "comportamiento", tenemos evolución de mercados, de compras online, tráfico en redes, *clicks* de usuarios, patrones de navegación en la Web, búsquedas en la Web, "likes" en Facebook, dinámica de redes sociales, etc. Estos datos se analizan permanentemente para predecir el comportamiento económico, mejorar máquinas de búsqueda, ofrecer mejores interfaces, estudiar comportamientos sociales, etc.

Tal cantidad de datos genera inmensas posibilidades de mejorar muchos aspectos de nuestra vida, y ya lo han hecho en gran medida. Dudo que quienes nacieron en la era de los buscadores puedan siquiera imaginar cómo era vivir cuando esa fuente instantánea e inagotable de información no estaba disponible. Mucho más está por venir también. Por ejemplo, la posibilidad de descubrir efectos de medicamentos no porque algún científico se ilumine y lo pruebe sino porque surja automáticamente del análisis y correlación de miles de historias clínicas, la posibilidad de que el automóvil se conduzca solo a través de una ciudad desconocida mediante interactuar con dispositivos inteligentes de la ciudad que lo guíen con información extraída automáticamente de la Web, y muchos otros ejemplos que hace una década eran ciencia ficción hoy están a la vuelta de la esquina.

Con las grandes posibilidades vienen los grandes desafíos. En cada una de las áreas mencionadas, hemos aprendido que es mucho más fácil generar datos que darles un uso útil. La humanidad genera unos cuantos zettabytes de datos al año, de acuerdo a un estudio de 2011 de la International Data Corporation (un zettabyte son 2^{70} bytes, aproximadamente 10^{21}), y el volumen crece a más del doble por año, más rápido que la Ley de Moore, que gobierna (en el mejor de los casos) el crecimiento de las capacidades de hardware. El libro "Algorithms and Data Structu-

res for External Memory" de Jeff Vitter comienza con una sentencia que resume la situación: *The world is drowning in data!* El término *data deluge*, o la inundación en datos, se ha puesto de moda para indicar que tenemos muchos más datos de los que podemos manejar.

Por lo pronto, no parece que nuestras capacidades de mero almacenamiento de datos estén aún comprometidas. Pero como decía, el problema no es realmente almacenar los datos sino darles algún uso. Incluso transmitirlos puede ser un problema: la compañía BGI, que secuencia datos en China, les envía a sus clientes las secuencias en una cinta por correo normal, ya que Internet no tiene el ancho de banda necesario para transmitirlas más rápidamente. ¿Qué esperar entonces de nuestras capacidades de procesar estos datos para obtener información útil? Realizar minería de textos o de datos en general para encontrar patrones relevantes, recorrer grafos para detectar comunidades de interés, buscar similitudes en secuencias genómicas, etc. son procesos que requieren ejecutar sofisticados algoritmos que recorran los datos, una o varias veces, con patrones de recorrido impredecibles. ¿Podemos realizar este procesamiento sobre datos almacenados en memoria secundaria (discos, SSDs) o terciaria (cintas)?

Lamentablemente, no todos los componentes del hardware mejoran según la Ley de Moore (que establece que la capacidad y/o la velocidad se duplica cada dos años aproximadamente). Lo hace la capacidad de la memoria externa (secundaria y terciaria), pero no su velocidad. Desde hace unos años, ya no lo hace la velocidad de los procesadores, aunque a cambio lo ha empezado a hacer la cantidad de procesadores en la CPU. La memoria interna (RAM) crece en capacidad, pero no mucho en velocidad, aunque aparecen cada vez más y mayores memorias caché rápidas entre la CPU y la memoria RAM. La diferencia entre procesar datos en memoria interna o externa es cada vez mayor, lo que ha hecho atractivo usar *clusters* de computadores para simular una gran memoria RAM.

LA REPETITIVIDAD: NUESTRA TABLA DE SALVACIÓN

Muchas de las secuencias, fotos, series de tiempo, etc. de las fuentes más prolíficas de datos son altamente *repetitivas*. Esto significa que es posible obtener cada nuevo elemento mediante combinar unas pocas partes de elementos vistos previamente. La repetitividad parece llamada a ser nuestra tabla de salvación frente al diluvio de datos. Un buen ejemplo son las colecciones de genomas. Existen muchas especies distintas, pero tampoco tantas. Las grandes colecciones de genomas se forman secuenciando muchos individuos de la misma especie. Dos genomas de individuos de la misma especie difieren en un porcentaje muy bajo, cercano al 0.1%. Otro ejemplo son los sistemas de versiones (de software, de documentos, etc.), con el caso públicamente disponible de *Wikipedia* y la *Internet Wayback Machine* (que almacena copias de páginas web a través del tiempo). Otra son las publicaciones de series de tiempo, financieras, climatológicas, etc., que en muchos casos tienen periodicidad con pocas variaciones. Los datos astronómicos suelen incluir "fotos" de la misma zona del cielo en distintos momentos

Se conocen buenas técnicas para tratar con datos muy repetitivos. Los sistemas que manejan versionamiento en edición colaborativa, como *CVS*, *Subversion*, y otros, almacenan solamente las diferencias de versiones menores con respecto a la versión anterior. Compresores genéricos como la familia Lempel-Ziv buscan, en el texto ya visto, copias del nuevo texto que deben comprimir. Elementos de esta familia como *p7zip* funcionan muy bien en colecciones altamente repetitivas, donde los compresores clásicos fallan en aprovechar la repetitividad.

Pero nuevamente estamos hablando de almacenar los datos eficientemente, permitiendo a

lo sumo recrear una versión específica. Si queremos tratar estas grandes colecciones repetitivas en forma eficiente, debemos ser capaces de *operar sobre los datos directamente en forma comprimida*, sin necesidad de descomprimirlos previamente.

ESTRUCTURAS DE DATOS COMPACTAS

Aquí entran en juego las *estructuras de datos compactas*, que combinan compresión y teoría de la información con estructuras de datos, de modo de representar los datos usando espacio cercano a lo que usaría un compresor, pero aún permitiendo navegar, operar y buscar en los datos sin descomprimirlos. En este aspecto van más allá de la compresión, que busca solamente representar los datos en menos espacio de modo de poder recuperarlos, y generalmente requiere descomprimirlos del todo antes de poder usarlos.

Las estructuras de datos compactas se han venido desarrollando cada vez con más fuerza desde fines de los ochenta, y hoy permiten representar una amplia gama de estructuras en espacio reducido: árboles, grafos, textos, grillas de puntos, etc. Pueden llegar a usar, en la práctica, treinta veces menos espacio que la estructura clásica correspondiente, y si eso permite que los datos que normalmente se almacenarían en memoria externa quepan en RAM, entonces la ganancia en tiempo de procesamiento es de órdenes de magnitud, incluso cuando manejar las estructuras compactas requiera bastantes más operaciones que las clásicas. En el caso de una memoria RAM distribuida en *clusters*, las estructuras compactas permiten reducir el número de computadores necesarios, su tiempo de comunicación, y su costo de hardware y de energía.

El desarrollo de estructuras de datos compactas específicas para explotar la repetitividad de los datos es bastante más reciente, y tiene todavía mucho terreno por explorar. Para ejemplificar el



tipo de problemas que enfrentamos al explotar la repetitividad, describiré el desarrollo de índices para colecciones de texto repetitivas, uno de los temas que se abordan en mi actual proyecto Fondecyt. Lo haré maximizando la intuición a través de ejemplos, evitando demostraciones o argumentos complejos.

COLECCIONES REPETITIVAS DE TEXTO

Para permitir búsquedas de diversa complejidad en colecciones de secuencias (por ejemplo secuencias de ADN), los índices favoritos son los *árboles de sufijos*, inventados en 1973 por Peter Weiner. Considere una colección formada por textos T_1, T_2, \dots, T_d , sobre un alfabeto Σ . Usaremos un símbolo extra, \$, para terminar los textos y los concatenaremos en un único texto $T[1,n] = T_1 \$ T_2 \$ \dots T_d \$$. Los árboles de sufijos se construyen sobre este texto concatenado T . Un *sufijo* de T es cualquier substring de T que termina en un \$. El *trie de sufijos* de T se obtiene insertando todos sus sufijos en un árbol digital, de modo que cualquier sufijo $T[i..]$ se puede leer recorriendo el árbol desde la raíz hasta la hoja asociada al sufijo que empieza en la posición "i" y concatenando las letras que rotulan las aristas.

El árbol de sufijos se obtiene compactando los caminos "unarios" (es decir, donde cada nodo tiene un único hijo) en el trie de sufijos. Ahora

cada arista está rotulada con un string y los strings que rotulan las aristas de cada nodo a sus hijos difieren en su primera letra. Como tiene n hojas y todo nodo interno tiene al menos dos hijos, el árbol de sufijos tiene menos de $2n$ nodos. También se puede construir en tiempo lineal, $O(n)$. La **Figura 1** muestra un árbol de sufijos para un solo string.

Podríamos dedicar muchas páginas a mostrar las virtudes y potencialidades del árbol de sufijos, pero nos contentaremos con un ejemplo simple e ilustrativo. Considere un problema de análisis de textos (para determinar autoría o plagio en textos de lenguaje natural, para buscar secuencias repetidas de ADN, etc.): encuentre el substring más largo de T que aparece dos veces. En un árbol de sufijos, esto corresponde simplemente a recorrer los nodos y encontrar el nodo interno que representa el string más largo, lo cual puede hacerse fácilmente en tiempo lineal (en nuestro ejemplo, el resultado es "ATA"). Sumado a que este árbol se puede construir en tiempo lineal, tenemos que este problema se puede resolver en tiempo $O(n)$. Pues bien, antes de que existieran los árboles de sufijos, incluso el legendario Donald Knuth creía que tal cosa no era posible. Esto ilustra la facilidad con la que el árbol de sufijos permite resolver problemas muy complejos.

Asimismo, podemos usar el árbol de sufijos para buscar las ocurrencias de un nuevo patrón, $P[1,m]$, en T . Simplemente hay que bajar por las letras de P hasta que (1) no se pueda continuar, en cuyo caso P no aparece en T ; (2) lleguemos

a una hoja, en cuyo caso hay que comparar el resto de P con esa posición del texto, para ver si P aparece allí o no aparece; o (3) se nos acaben las letras de P , en cuyo caso el nodo en que estamos, o donde nos lleva la arista en que estamos, corresponde a todas las ocurrencias de P en T , una por hoja (y las posiciones donde ocurre P son los valores asociados a las hojas). En nuestro ejemplo, tenemos búsquedas de tipo (1), (2) o (3) cuando buscamos "TT", "CGA", o "AT", respectivamente. El árbol de sufijos realiza esta búsqueda en tiempo $O(m)$, independientemente del largo del texto, lo que es notable.

Esta poderosa herramienta tiene, sin embargo, un serio problema: ocupa mucho espacio. En una implementación típica, el árbol de sufijos puede ocupar $20n$ bytes. Si tenemos un genoma humano (unos 3 mil millones de bases), el árbol de sufijos requiere unos 60GB de memoria. Además los recorridos en el árbol siguen caminos impredecibles (no todos los problemas se resuelven con recorridos top-down, por lo que el árbol debe mantenerse en memoria para que los recorridos funcionen eficientemente). Si 60 GB de RAM parece una exigencia considerable, piense en la razonable aspiración de manejar una colección de 1000 genomas: requeriría 60 terabytes de RAM!

Sin embargo, la repetitividad es un poderoso aliado para reducir el espacio de los árboles de sufijos de colecciones repetitivas. La **Figura 2** muestra una colección de varios strings similares.

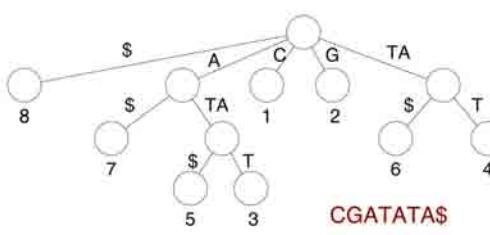


FIGURA 1.
ÁRBOL DE SUFIJOS PARA UN SOLO STRING. COMO LOS STRINGS QUE LLEGAN A LAS HOJAS SON MUY LARGOS, ALMACENAMOS SÓLO SU PRIMERA LETRA.

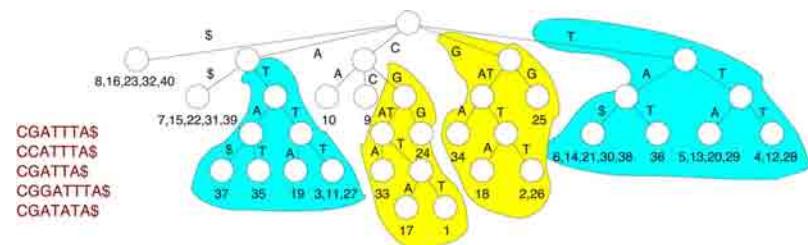


FIGURA 2.
ÁRBOL DE SUFIJOS PARA UNA COLECCIÓN DE VARIOS STRINGS SIMILARES.

Hemos mostrado, en celeste y amarillo, dos zonas del árbol que son topológicamente idénticas. La repetitividad en los textos induce zonas topológicamente idénticas en el árbol de sufijos. Veremos luego cómo explotar esa repetitividad. Veamos, por lo pronto, lo que ocurre con las posiciones asociadas a las hojas del árbol, como se muestra en la **Figura 3**.

Las áreas del mismo color son iguales, excepto que sus valores difieren en una unidad (estas áreas se traslanan, como puede verse). Nuevamente, veremos luego cómo explotar este hecho: *La repetitividad en los textos induce largas áreas en las posiciones del árbol de sufijos cuyos valores difieren en una unidad.*

La primera representación compacta de árboles de sufijos fue propuesta por Kunihiko Sadakane en 2007. Consta de tres partes: la topología compacta, la secuencia de posiciones (que se llama *arreglo de sufijos*) y otro arreglo de prefijos comunes más largos (*LCP, por longest common prefix*). Este arreglo indica el largo del prefijo común que comparten dos hojas consecutivas del árbol. Estos tres componentes son suficientes para simular todas las operaciones en un árbol de sufijos. Ni siquiera se necesita el texto. Incluso se puede extraer de la estructura cualquier substring del texto que se desee, ¡por lo que el texto puede borrarse!

La **Figura 4** muestra el arreglo LCP de nuestra colección. Como puede verse, persiste el mismo tipo de repetitividad que en el arreglo de sufijos, a veces con excepción del primer elemento de las áreas. La repetitividad en los textos induce largas áreas en el arreglo LCP cuyos valores difieren en una unidad.

El árbol de sufijos compacto de Sadakane obtiene un espacio cercano a unos $10n$ bits para el caso de ADN. Esto es 1,25 bytes por símbolo, muy inferior a los 20 de las implementaciones clásicas. Un genoma humano cabe en menos de 4GB, que es muy razonable para una memoria principal. Sin embargo, para nuestro razonable objetivo de 1000 genomas, una memoria principal de 4TB es todavía de muy alta gama. Mostraremos cómo se ha logrado reducir este espacio a unos $2n$ bits para el caso de las colecciones repetitivas, con lo cual los 1000 genomas humanos cabrían en menos de 700GB.

queño. La idea es compactar cada uno de los componentes utilizando distintas herramientas. Una idea genérica que es útil es la compresión basada en *gramáticas libres del contexto*. Estas representan una secuencia de símbolos como una serie de *reglas*, donde un símbolo *no terminal* se reescribe como una secuencia de símbolos *terminales* (los de la secuencia) y otros no terminales. Los no terminales, a su vez, se reescriben usando las reglas, hasta que la secuencia contiene sólo símbolos terminales. Las secuencias repetitivas pueden así representarse como la cadena que se deriva de un cierto símbolo *no terminal*, que se llama *símbolo inicial*, y la gramática puede ser mucho menor que la secuencia.

Para ilustrar esta idea, aplíquemósela a la versión *diferencial* de nuestro arreglo de sufijos. En la versión diferencial, cada elemento se escribe como la diferencia con el anterior. Esto hace que las zonas largas que se repetían en otra parte con una diferencia de una unidad, ahora se convierten en zonas idénticas.

Nuestro arreglo de sufijos diferencial se ve así tal como se muestra en la **Figura 5**.

Como se puede ver, las repeticiones se han conservado (excepto por los primeros símbolos). Gracias a estas repeticiones, el arreglo se puede representar con una gramática bastante más

ÁRBOLES DE SUFIJOS PARA COLECCIONES REPETITIVAS

Mostraremos cómo aprovechamos la repetitividad para obtener un árbol de sufijos tan pe-



FIGURA 3.
ARREGLO DE SUFIJOS PARA NUESTRA COLECCIÓN.



FIGURA 4.
ARREGLO LCP PARA NUESTRA COLECCIÓN.

corta, de 33 símbolos (contando los largos de los lados derechos de las reglas, que es cercano al costo real de almacenarla) en vez de 40, donde S es el símbolo inicial:

$$\begin{aligned} S &\rightarrow 8 \ B \ -33 \ E \ -2 \ C \ D \ -17 \ -1 \ 24 \ C \ 23 \ 10 \ C \ 24 \ -1 \ -19 \ E \ -31 \ A \ -25 \ D \\ A &\rightarrow 8 \ 7 \ 9 \\ B &\rightarrow A \ 8 \\ C &\rightarrow -16 \ -16 \\ D &\rightarrow 8 \ 16 \\ E &\rightarrow B \ -2 \end{aligned}$$

La diferencia sería más impresionante en un ejemplo más largo, claro. Veremos ahora cómo representar cada componente del árbol de sufijos compacto.

TOPOLOGÍA DEL ÁRBOL: la representación compacta clásica de un árbol de N nodos usa $2N$ bits: se recorre el árbol en preorden abriendo un paréntesis al llegar a un nodo y cerrándolo al dejarlo definitivamente. Esta secuencia se parte en bloques, y se construye una jerarquía sobre los bloques. Para cada nodo de esta jerarquía se almacenan datos como el exceso total (suma de paréntesis que abren menos paréntesis que cierran), el mínimo y máximo exceso al recorrer el área de izquierda a derecha, etc. Esta jerarquía es suficiente para permitir navegar los paréntesis eficientemente, como si tuviéramos el árbol. La secuencia de paréntesis correspondiente a nues-

tro árbol de sufijos sobre la colección repetitiva es lo que se muestra en la **Figura 6**.

Como puede verse, los subárboles idénticos se traducen en substrings idénticos en la secuencia de paréntesis. Podemos entonces representarla usando una gramática, y asociar la información sobre excesos a los no terminales de la gramática.

Esto produce una representación mucho menor que los $2N$ bits cuando la secuencia es repetitiva (nótese que nuestro árbol de sufijos tiene hasta $N=2n$ nodos, por lo que puede ocupar hasta $4n$ bits). Con una representación de gramáticas, el espacio para colecciones repetitivas ocupa unos n bits. Una gramática para generar nuestra secuencia de paréntesis podría ser como sigue:

$$\begin{aligned} S &\rightarrow E \ E \ A \) \ C \ B \) \ B \ A) \\ A &\rightarrow (\ D \ D) \\ B &\rightarrow (\ E \ D) () \\ C &\rightarrow E \ () \\ D &\rightarrow C \\ E &\rightarrow (\) \end{aligned}$$

Existen incluso representaciones de árboles de sufijos que no utilizan en absoluto los paréntesis,

sino que los deducen de los otros componentes, especialmente del arreglo LCP, el que se almacena en forma de gramática para agilizar las operaciones. Para secuencias muy repetitivas, pueden llegar a necesitar sólo $1.5n$ bits en total, pudiendo almacenar 1000 genomas en casi 500GB. Sin embargo, son también varios órdenes de magnitud más lentos.

Exploramos la vía de eliminar del todo la topología del árbol con mi exalumno memorista Andrés Abeliuk en 2012, obteniendo el menor árbol de sufijos hasta el momento. Con mi alumno de Doctorado Alberto Ordóñez, de la Universidad da Coruña, exploramos la variante de usar gramáticas para comprimir la topología en 2014, resultando en una estructura algo mayor, pero hasta mil veces más rápida que la que prescinde completamente del árbol.

ARREGLO DE SUFIJOS: es posible obtener una representación compacta del arreglo de sufijos usando gramáticas, como vimos en el ejemplo. Probamos esta idea en 2007 con mi alumno de Doctorado Rodrigo González. Si bien el acceso al arreglo de sufijos es muy rápido, la compresión que se alcanza no es competitiva con otras técnicas. Una que funciona mucho mejor, propuesta por Sadakane en 2000, es reemplazar el arreglo de sufijos $A[1,n]$ por otro que se llama $\Psi[1,n]$, que contiene en $\Psi[i]$ la posición j donde



FIGURA 5.
ARREGLO DE SUFIJOS DIFERENCIAL PARA NUESTRA COLECCIÓN.



FIGURA 6.
REPRESENTACIÓN CON PARÉNTESIS DE LA TOPOLOGÍA DE NUESTRO ÁRBOL DE SUFIJOS.

$A[j]=A[i]+1$ (excepto si $A[i]=n$, en cuyo caso contiene la posición j donde $A[j]=1$). Requeriría mucho espacio explicar cómo este nuevo arreglo puede reemplazar al arreglo de sufijos. En cambio, nos centraremos en sus propiedades.

La Figura 7, muestra el arreglo Ψ de nuestro ejemplo.

Hay dos cosas importantes que resaltar: primero, el arreglo es creciente en la zona de los sufijos que empiezan con cada letra (las letras iniciales se indican abajo). Con un poco de cuidado eso puede valer para la zona de los \$ también. Esto vale para cualquier tipo de texto y permite codificar el arreglo Ψ en un espacio cercano al texto comprimido con un compresor simple, tipo Huffman. Esto representa mucho menos espacio que el arreglo de sufijos original.

El segundo hecho es relevante para las colecciones repetitivas: las zonas coloreadas se convierten en áreas de Ψ donde el arreglo crece de a una unidad, y por lo tanto es muy fácil de codificar en muy poco espacio. Esto ocurre en todas las zonas de cada color excepto en una, donde los números son mayores. Nuevamente, no tenemos suficiente espacio para explicar las razones por las que esto ocurre. Con Veli Mäkinen y sus alumnos, de la University of Helsinki, publi-

camos en 2008 el Run-Length Compressed Suffix Array, que usa esta idea y se ha convertido en la principal estructura para utilizar en esta situación. En colecciones repetitivas, esta estructura usa unos $0.7n$ bits.

ARREGLO LCP: como mencionamos, el arreglo LCP se puede comprimir usando una gramática y eso es buena idea cuando no se representa la topología del árbol, pues se necesita accederlo muy frecuentemente. Cuando se tiene la topología, los accesos a LCP son mucho más esporádicos y se puede utilizar una representación más compacta. Sadakane propuso en 2007 utilizar a cambio el arreglo *PLCP* (Permuted LCP), donde $PLCP[i] = LCP[A[i]]$, es decir, almacena los mismos valores pero en orden de texto. Esto es conveniente porque, en ese orden, vale que $PLCP[i+1] \geq PLCP[i]-1$. Esto permite marcar los valores $PLCP[i]+2i$ (que es una secuencia estrictamente creciente) como 1s en un vector de $2n$ bits, y recuperar eficientemente la posición del i -ésimo 1 del vector para encontrar el valor de $PLCP[i]$. Además de esta notable reducción de espacio, se puede ganar aún más en las colecciones repetitivas. El arreglo *PLCP* y el vector de bits para nuestro ejemplo se muestran en la **Figura 8**.

Obsérvese cómo aparecen largas zonas decrecientes en una unidad en *PLCP*, que se traducen en largas zonas de 1s (y por lo tanto de 0s, ya que hay n de cada uno) en el vector de bits. Esta propiedad, que demostramos con Veli Mäkinen y Johannes Fischer, del KIT Alemania, en 2008, hace al vector de bits mucho menor que $2n$ bits en colecciones repetitivas, como confirmamos con mi exalumno Andrés Abeliuk: ¡se llega a usar unos $0.2n$ bits!

Sumando los n bits de los paréntesis, los $0.7n$ bits del arreglo de sufijos, y los $0.2n$ bits del *LCP*, tenemos los $2n$ bits mencionados. Sacando los paréntesis y agregando una compresión de gramáticas para *LCP*, obtenemos los $1.5n$ bits ya mencionados también. Cuando se incluye la topología del árbol, la estructura realiza las operaciones en microsegundos, mientras que cuando no, se llega a los milisegundos. Para poner estos resultados en contexto, la representación que usa $2n$ bits es aún diez veces más lenta que el árbol de sufijos compacto de Sadakane (pero también ocupa cinco veces menos espacio, y va ocupando menos a medida que la repetitividad aumenta). Un árbol de sufijos clásico ¡ocupa 320 veces más espacio que la representación usando paréntesis comprimidos con gramáticas!

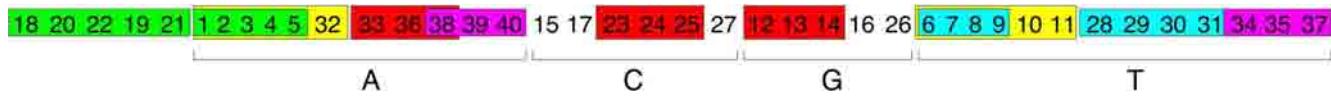


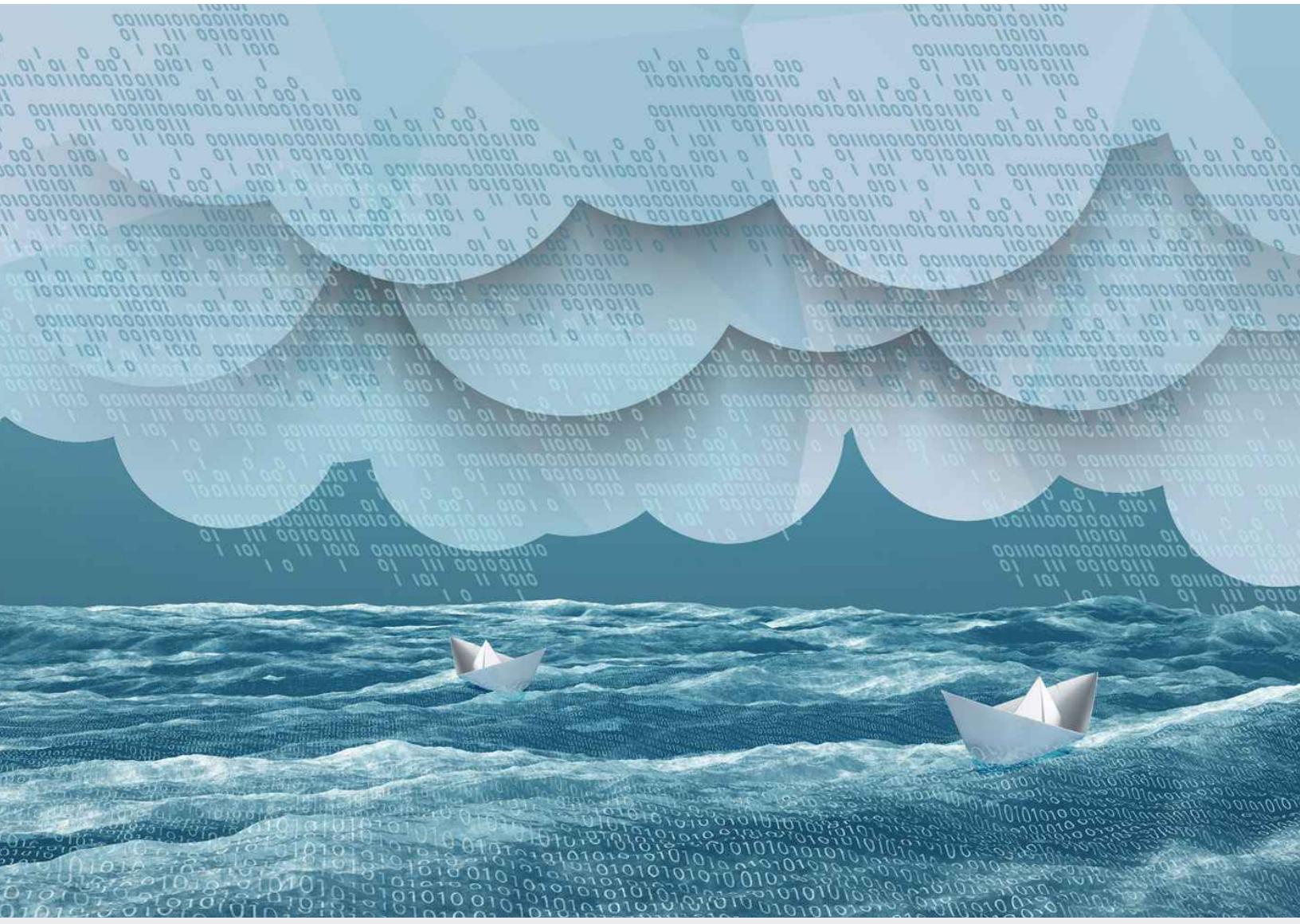
FIGURA 7.

ARREGLO Ψ DE NUESTRA COLECCIÓN.

5 4 3 2 1 0 0 0 1 0 3 2 1 0 0 0 4 3 2 1 0 0 0 2 1 4 3 2 1 0 0 0 1 0 3 2 1 0 0 0
0000001111110101001100001111010100000111110101000110000111110101001100001

FIGURA 8.

ARREGLO *PLCP* Y SU REPRESENTACIÓN COMO BITS PARA NUESTRA COLECCIÓN.

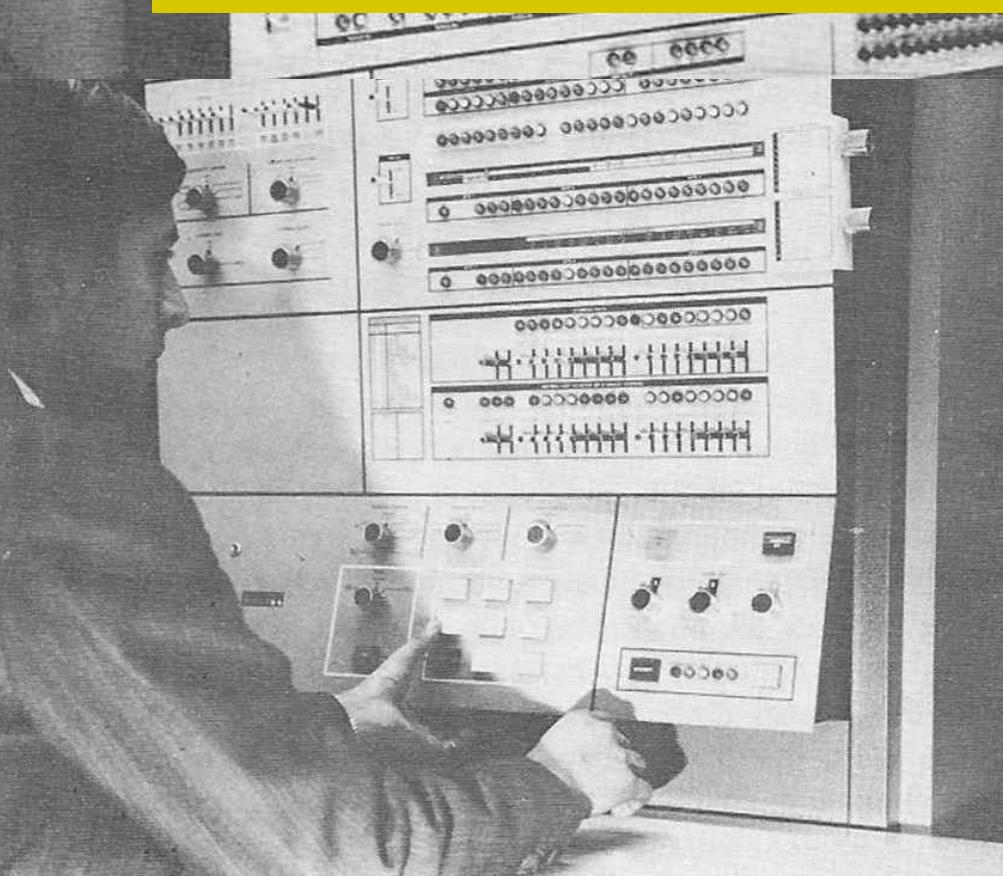


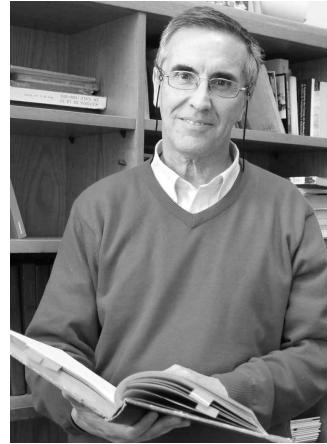
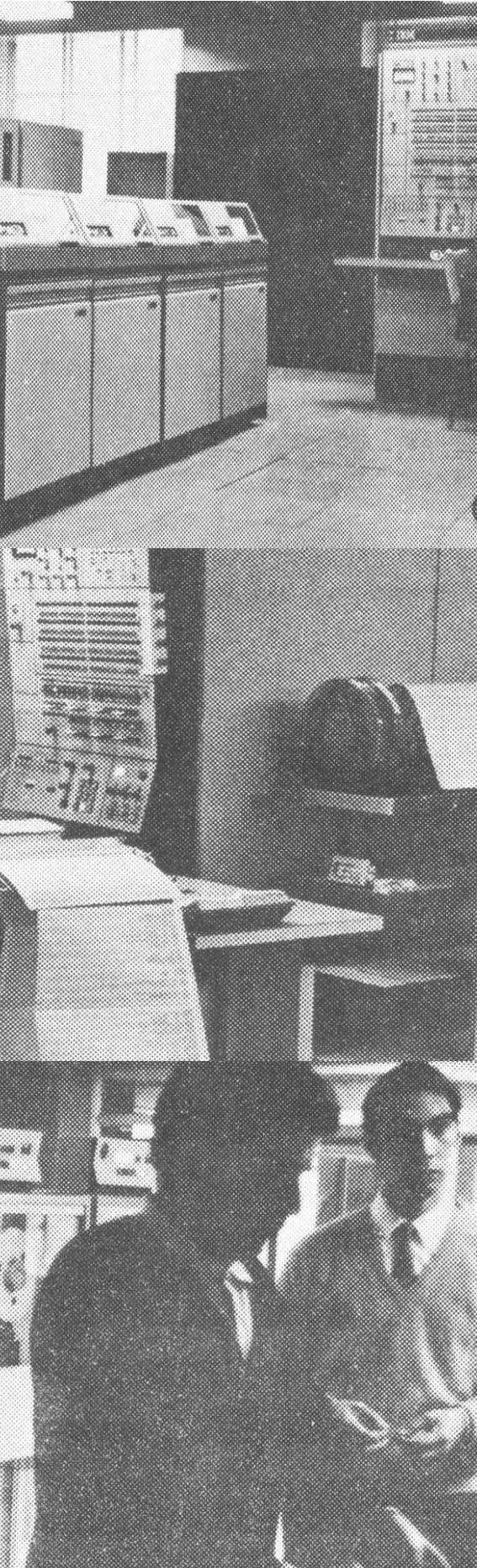
CONCLUSIONES

EXPLOTAR LA REPETITIVIDAD DE LOS DATOS PUEDE SER LA CLAVE PARA AFRONTAR EL DILUVIO DE DATOS QUE SE AVECINA RÁPIDAMENTE. EL ENTENDER CÓMO LA REPETITIVIDAD EN LOS DATOS SE MANIFIESTA EN LAS DISTINTAS ESTRUCTURAS DE DATOS QUE USAMOS PARA ACCEDER EFICIENTEMENTE A ELLOS PERMITE DISEÑAR VARIANTES DE ESTAS ESTRUCTURAS QUE REDUCEN SU ESPACIO EN ÓRDENES DE MAGNITUD. MOSTRAMOS CON CIERTO DETALLE EL CASO DE LOS ÁRBOLES DE SUFIJOS, UNA HERRAMIENTA MUY IMPORTANTE PARA EL ANÁLISIS DE SECUENCIAS. EL SEGUIR EL MISMO CAMINO CON OTRAS ESTRUCTURAS Y OTROS TIPOS DE DATOS REQUIERE UN TRABAJO ALGORÍTMICO MUY ESTIMULANTE EN UN ÁREA QUE SE CONSTRUYE SOBRE UNA LARGA TRADICIÓN DE INVESTIGACIÓN EN ALGORÍTMOS Y ESTRUCTURAS DE DATOS Y TEORÍA DE LA INFORMACIÓN, PERO QUE A LA VEZ ESTÁ CASI INEXPLORADA, ESPERANDO LA CONTRIBUCIÓN DE NUEVOS INVESTIGADORES. ■

EMPRESA NACIONAL DE COMPUTACIÓN E INFORMÁTICA (ECOM), 1971-1973

"Empresa Nacional de Computación e Informática" (ECOM) fue el nombre que recibió en diciembre de 1970 la "Empresa de Servicio de Computación" (EMCO) creada anteriormente en septiembre de 1968 como filial de CORFO, Endesa y Entel. En el período 1971-1973 ECOM amplió su cobertura más allá de los servicios y la capacitación, desarrollando investigación aplicada que permitió transitar desde el desarrollo de proyectos "a la medida" hacia la construcción de paquetes de software genéricos y la participación en la realización de proyectos de gran envergadura e impacto nacional (Censo, RUN, Syncro).





JUAN ÁLVAREZ

Académico Departamento de Ciencias de la Computación, Universidad de Chile. Master of Mathematics (Computer Science), University of Waterloo. Ingeniero de Ejecución en Procesamiento de la Información, Universidad de Chile. Junto a su labor como docente, trabaja en reconstruir la Historia de la Computación en Chile.

jalvarez@dcc.uchile.cl

INTRODUCCIÓN

La Empresa de Servicio de Computación (EMCO), fue creada en septiembre de 1968 con el propósito de: 1) proporcionar servicios de procesamiento de datos a entidades del Estado; 2) difundir el uso de sistemas de procesamiento de datos y de equipos electrónicos; y 3) preparar personal técnico en análisis y programación de sistemas para los Servicios del Estado [1]. En abril de 1970, un balance de la operación [2] mostraba que EMCO:

- *Cubre el servicio del 30% de la demanda con sólo un 17% del gasto total del país.*
- *Ofrece el 81% del servicio de la demanda del sector público, con una inversión del 55% del total del sector.*
- *Atiende a 41 instituciones del Estado que se proyectan a 50 para fines de 1970.*
- *De 40 funcionarios en diciembre de 1968, pasó a 120 en diciembre de 1969 y se proyecta contar con 230 en 1970.*
- *Cuenta con 3 computadores propios con US\$84.500 de gasto mensual.*

Considerando que las actividades proyectadas para la empresa superarían las de un servicio de computación, el 24 de diciembre de 1970 EMCO cambió su nombre a "Empresa Nacional de Computación e Informática" (ECOM). En otras palabras se eliminó la palabra "servicio" y se incorporó "nacional" en consistencia con su pretensión de operar en todo el país. "Informática", término nuevo en la época, se agregó para enmarcar el tipo de actividades de la empresa: "La informática se dedica a la investigación

de las propiedades, el comportamiento y las leyes que gobiernan los procesos de transferencia y la tecnología para el procesamiento de la información con el objeto de optimizar su acceso y uso" [3].

A continuación, se presentan los principales recursos y proyectos realizados por ECOM en el corto e intenso período comprendido entre diciembre de 1970 y septiembre de 1973.

ESTRUCTURA Y PERSONAL

En 1970, una vez que el Dr. Salvador Allende asumió la presidencia del país, se designó a los ejecutivos de ECOM: Raimundo Beca, economista de la Universidad Católica, como gerente general y como Presidente del Directorio; Héctor Hugo Segovia, ingeniero industrial de la Universidad de Chile, como gerente de Ingeniería; Gonzalo Vargas, ingeniero industrial de la Universidad de Chile, como gerente de operaciones. Los departamentos de la Gerencia de Ingeniería continuaron a cargo de René Barros (Análisis), Jorge Gutiérrez (Programación), Germán Munita (Investigación y Desarrollo) y Mario Pardo (Capacitación). Posteriormente, se creó la Gerencia de Planificación y Desarrollo a cargo de Héctor Hugo Segovia y en el Departamento de Investigación y Desarrollo asumió Isaquino Benadof.

Respecto del personal, a fines de 1971 se contaba con 253 personas (que se proyectaban a 373 para 1972), distribuidos en 12 directivos (15 para

1972), 67 profesionales/técnicos (98), 41 administrativos (50), 109 calificados (178) y 24 no calificados (32) [4].

COMPUTADORES

En 1971 ECOM disponía de tres computadores IBM/360 [1] y las estadísticas de uso, en comparación con el año 1970, fueron las siguientes [5]:

- *El IBM/360 modelo 40 con 256K de memoria se usó 6036 horas (6360 en 1970).*
- *El IBM/360-50 con 384K de memoria se utilizó 5429 horas (3988 en 1970).*
- *El IBM/360-40 con 128K (instalado en Endesa) se usó 4308 horas (3213 en 1970).*

En 1972 ECOM adquirió un computador Burroughs 3500 (**Figura 1**) orientado al procesamiento de datos de tipo administrativo del sector financiero del Gobierno. El total del monto de la inversión fue de E\$8.200.000 y fue pagado por CORFO. La configuración, similar a la del computador Burroughs 3500 de la Universidad Católica con quien se estableció un convenio de respaldo mutuo, era la siguiente [6]:

- *Procesador central B3501 con 120 Kilobytes de memoria.*
- *Consola (SPO) B9340.*
- *Unidad electrónica de 20.000KD.*
- *Unidad electrónica de 40.000KD.*
- *Impresora B9243.*
- *2 unidades de cinta B9381 para densidad de grabación de 800 bytes por pulgada.*
- *4 unidades de cinta B9393 para 1600 bytes por pulgada.*
- *Unidad lectora de tarjetas B9112.*

Por otra parte, a fines de 1973 iniciaría su operación el sistema IRIS-80 adquirido por ECOM a la compañía francesa CII (Compagnie Internationale pour l'Informatique) en US\$3.100.000 (**Figura 2**). El computador "será instalado en un edificio que está construyendo la empresa a los pies del cerro Manquehue" y "pasaría a ser en el mediano plazo el núcleo de una red nacional de



FIGURA 1.
COMPUTADOR BURROUGHS 3500 DE ECOM.

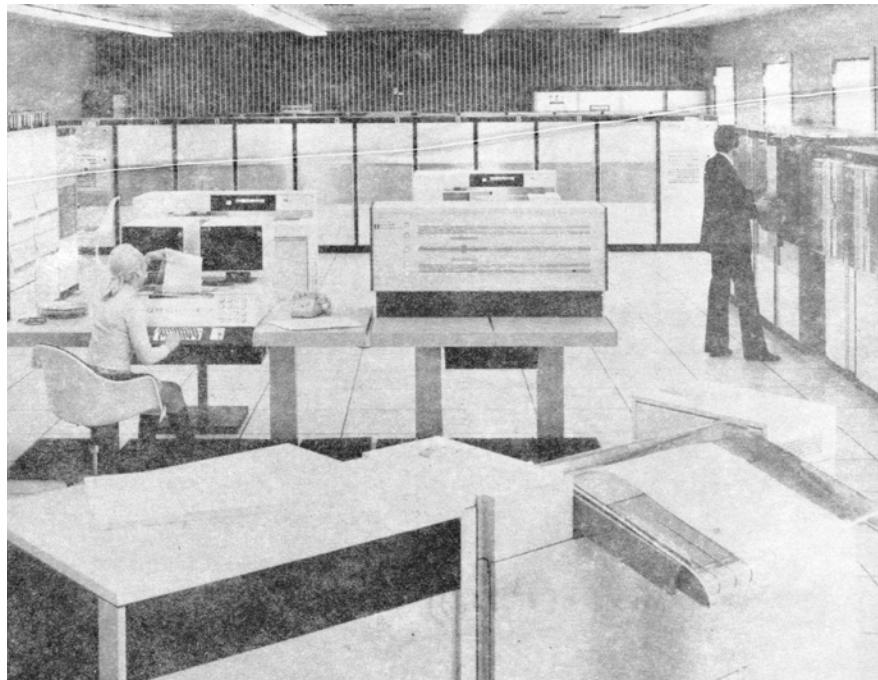


FIGURA 2.
COMPUTADOR IRIS-80 DE CII QUE LLEGARÍA A ECOM A FINES DE 1973.

computadores” [7]. Además se adquirió un computador IRIS-60 en US\$1.240.000 destinado a la Zona de Concepción (**Tabla 1**).

Adicionalmente se contaría con minicomputadores MITRA-15 de CII: dos en Santiago y uno en Concepción. Estarían equipados con lectora de tarjetas (de 300 tarjetas por minuto) e impresora (de 200 líneas por minuto) y serían utilizados como procesadores independientes y como terminales inteligentes. En el caso de Concepción se conectaría también como terminal al sistema IRIS-80 de Santiago.

Con la llegada del nuevo computador, uno de los ya existentes en Santiago sería trasladado a Valparaíso principalmente para las aplicaciones de la Superintendencia de Aduanas, Emporchi y Sudamericana de Vapores. Por otra parte, se adquiriría un computador IBM/370-145 que sería propiedad de la Universidad Técnica del Estado (UTE, hoy USACH) y ECOM cancelaría como arriendo aproximadamente un 77% del valor del equipo (aproximadamente 1,1 millones de dólares). La UTE entregaría a ECOM la administración y comercialización del tiempo de computador y la Universidad tendría derecho a uso liberado hasta por un 23% de su capacidad horaria [4].

CAPACITACIÓN Y SERVICIOS

En consistencia con sus propósitos fundacionales, ECOM continuó ofreciendo servicios de capacitación y procesamiento de datos a diferentes instituciones del Estado. En el área de capacitación, y en comparación con años anteriores (**Tabla 2**), creció notoriamente la formación de interlocutores, concebidos como ejecutivos medios que pueden intermediar entre los usuarios y los técnicos especialistas[5][4].

CURSO	1969	1970	1971	1972
Programación	30	57	38	75
Análisis	23	73	53	75
Interlocución		75	155	120
Total	53	205	246	270

TABLA 2.
ESTADÍSTICAS DE CURSOS DE CAPACITACIÓN.

UNIDAD	VELOCIDAD	IRIS-80	IRIS-60
Unidad Central de Proceso	1.200.00 instrucciones por segundo	2	1
Memoria	Ciclo de 650 nanosegundos	768K	192K
Procesador multiplexor de E/S	42000 kilobytes por segundo	2	1
Unidad de Cinta de 1600bpi	120 kilobytes por segundo	10	4
Unidad de Cinta de 800bpi	60 kilobytes por segundo	2	1
Disco magnético	Tiempo acceso: 17-50 mili segundos	600M bytes	150M bytes
Impresora	1200 líneas por minuto	4	2
Lectora de tarjetas	1000 tarjetas por minuto	2	1
Perforadora de tarjetas	200 tarjetas por minuto	1	1
Terminal teletipo	10 caracteres por segundo		16

TABLA 1.
CONFIGURACIONES DE LOS COMPUTADORES IRIS-80 E IRIS-60.

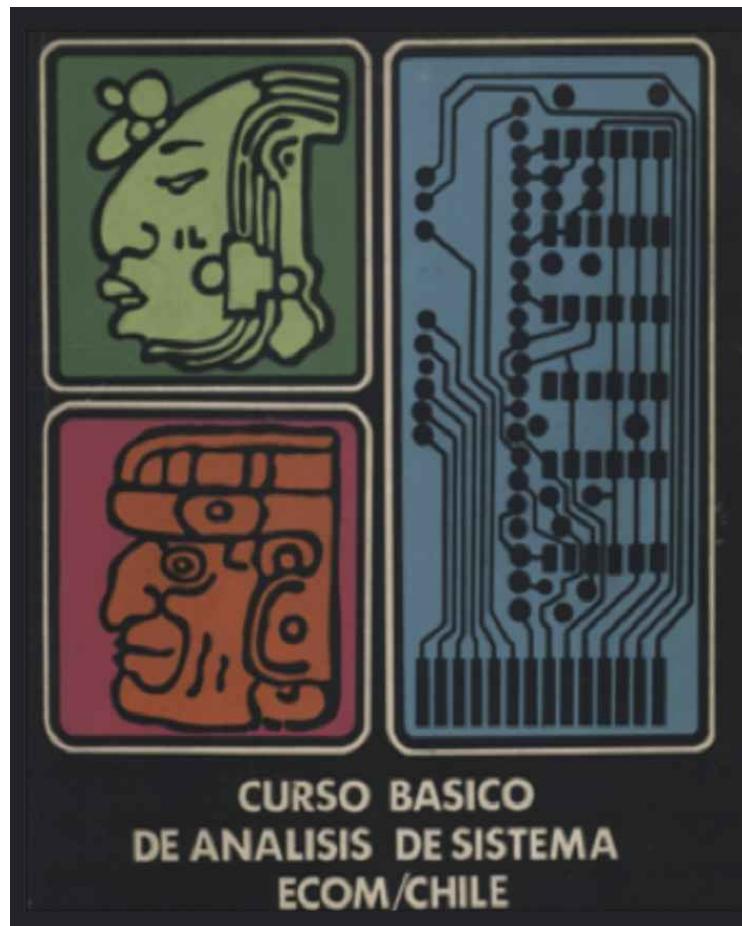


FIGURA 3.
PORTADA DE MATERIAL DE CURSO DE CAPACITACIÓN.

El curso básico de análisis de sistemas (**Figura 3**) duraba cinco meses a tiempo completo y dedicación exclusiva. El plan estaba basado en un curso del National Computing Centre de Gran Bretaña y comprendía las siguientes unidades: Fundamentos de Computación (cinco semanas), Introducción al Sistema 360 (dos semanas), Básico de Programación Cobol (cuatro semanas), Orientación administrativa (tres semanas) y Básico de Análisis de Sistemas (siete semanas). Los postulantes debían tener 25 o más años, ser egresados de la universidad y al menos dos años de trabajo en cargos de responsabilidad administrativa/financiera y/o en procesamiento electrónico de datos. Adicionalmente debían aprobar el sistema de selección de ECOM en base a entrevistas y un test de aptitudes [8].

Por otra parte, ECOM prestó servicios de tiempo de computador (en modalidad "Service" y "Block Time"), de perfo-verificación, y de asesoría en análisis de sistemas. El servicio de procesamiento de datos se extendió a más instituciones del Estado, siendo los principales usuarios Enap, El Teniente, Instituto Nacional de Estadísticas, Centro de Informaciones del Ministerio de Vivienda, Endesa, Caja de Previsión de Empleados Particulares, Sector Financiero del Gobierno, Ministerio de Justicia, Ministerio de Educación, Ferrocarriles del Estado, Servicio Médico Nacional de Empleados, Servicio de Seguro Social, Caja de Empleados Públicos, Registro Electoral, Oficina de Planificación Agrícola, Municipalidad de Providencia, Fondo de Extensión y Educación Sindical, Compañías del Cobre y usuarios de Valparaíso y Concepción [9].

CENSO Y PROYECTO RUN-RNP

En el período 1971-1973 ECOM participó en proyectos nacionales de gran envergadura. Al respecto, el martes 21 de abril de 1970 se realizó el XIV Censo de Población y el III de Vivienda. El diario La Nación tituló en su portada "El censo a las computadoras – Alimentado con 10 millones de tarjetas, cerebro electrónico analizará datos

CUADRO N. 3.-		POBLACIÓN TOTAL POR ÁREA URBANA Y RURAL Y SEXO, SEGUN GRUPOS QUINQUENALES Y AÑOS SIMPLES DE EDAD.									
GRUPOS QUINQUENALES DE EDAD	AÑOS SIMPLES Y	POBLACIÓN									
		TOTAL			ÁREA URBANA			ÁREA RURAL			
		TOTAL	HOMBRES	MUJERES	TOTAL	HOMBRES	MUJERES	TOTAL	HOMBRES	MUJERES	
TOTAL		8884768	4343512	4541256	6675137	3173323	3501814	2209631	1170189	1039442	.

FIGURA 4.

RESULTADOS GENERALES DEL CENSO 1970 PROCESADO POR ECOM.

del Censo 1970" informando que, gracias a que por primera vez se utilizaría un computador, el detalle se conocería en un plazo de dos a tres años [10]. Posteriormente, al presentar los resultados (**Figura 4**) el Instituto Nacional de Estadísticas expresó [11]:

Deseamos dejar constancia que el procedimiento computacional de los datos estuvo a cargo de la Empresa Nacional de Computación (ECOM). Una de las etapas más importantes de este proceso fue la confección de un programa de validación y asignación automática de datos (ASIG), el que permite la presentación de la información sin errores de consistencia estadística. Este programa fue el producto del trabajo conjunto de ambas instituciones y se elaboró sobre la base de criterios estadísticos, que aseguran la plena validez cuantitativa de la información que se presenta.

Por otra parte, ECOM participó activamente en el "Proyecto Rol Único Nacional y Registro Nacional de Población" realizado en conjunto con la Subsecretaría de Justicia, la Dirección de Presupuesto, el Registro Civil e Identificación y el Servicio de Impuestos Internos. La Comisión Técnica del Proyecto "fue creada a mediados de 1971, y tuvo como principal objetivo el formular las políticas y programas de trabajo para iniciar durante 1972 la creación de un Registro Nacional de Población (RNP) y la implantación de un número único de identificación para todos los habitantes del país (RUN)" [12].

Para alcanzar el normal funcionamiento de los sistemas computacionales y administrativos se estimó un período de aproximadamente cuatro años. ECOM participó con dos de los seis integrantes en la Comisión Técnica (Hugo Segovia y René Barros) y destinó un equipo de nueve personas para construir un proyecto piloto y diseñar e implementar los sistemas computacionales definitivos. Adicionalmente se contó con la asesoría de Pinchas Stern, experto de Naciones Unidas en Sistemas de Información.

Cabe señalar que el proyecto RUN-RNP necesitó investigación acerca del diseño de números de identificación con dígitos verificadores y respecto del procesamiento de consultas y de la recuperación de información utilizando y adaptando códigos SOUNDEX. El trabajo de investigación y los desarrollos realizados permitieron la publicación de artículos, reportes técnicos y presentaciones en seminarios nacionales [13] y congresos internacionales [14].

SYNCO

En noviembre de 1971 el Presidente Allende, después de una reunión con el cibernetico inglés Stafford Beer propiciada por el subgerente técnico de CORFO Fernando Flores, aprobó iniciar el proyecto Sync (inicialmente llamado Cybersyn) cuyo objetivo era desarrollar "Sistemas de Información y Control" de la producción de las



¡Nuestro amigo el Computador!

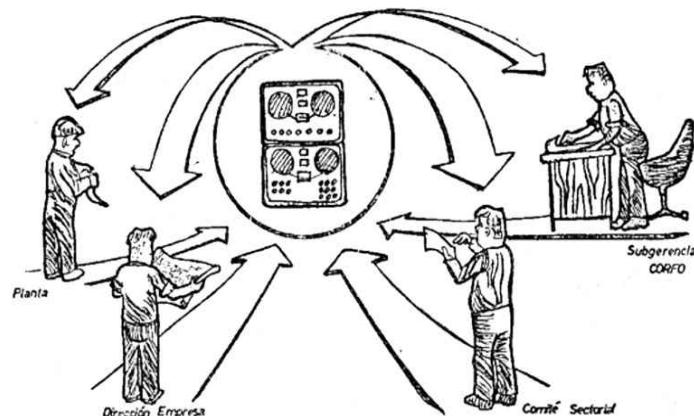


FIGURA 5.
DIFUSIÓN DEL COMPONENTE CYBERSTRIDE DEL PROYECTO SYNC.

empresas del área social [15]. El proyecto fue concebido en la Dirección Industrial de CORFO, involucró también al Instituto Tecnológico (INTEC) y a ECOM y contempló cuatro componentes [16]:

1. *Establecimiento de una Red de Comunicaciones (Cybernet) entre las principales empresas del Área Social y CORFO a través de un Sistema Télex.*
2. *El desarrollo de un Sistema Computacional para el control de la gestión operativa del Área Social en tiempo real (CYBERSTRIDE).*
3. *La simulación dinámica de sistemas, con el objetivo de apoyar las actividades de desarrollo y planificación del aparato industrial (CHECO).*
4. *La construcción de una Sala de Operaciones (Opsroom) con el objetivo de facilitar la toma de decisiones, a través de crear condiciones favorables para la trasmisión y uso de la información que generan los sistemas anteriores.*

En un trabajo colaborativo con un grupo de investigadores en Inglaterra, ECOM desarrolló Cyberstride y destinó algunos investigadores a CHECO. Cyberstride se implementó inicialmente en el computador IBM/360-50, pero debido a exigencias de alta disponibilidad fue portado, en muy poco tiempo y no sin dificultades, por Isaquino Benadof y su equipo al computador Burroughs 3500 donde finalmente estuvo disponible en los últimos meses de 1972, realizándose una demostración para el presidente Allende el día 30 de diciembre [17].

El propósito de Cyberstride se difundió a los usuarios de las industrias y de CORFO a través de un dibujo (**Figura 5**) y una descripción sencilla: "Lo que hará el computador es recolectar cada día la información e informar a los interesados cuando éste ocurriendo alguna anormalidad, anormalidad que es preestablecida antes de enviar la información al computador" [18]. Las ventajas se resumían precisando que la utilización del sistema:

1. *Elimina los informes, cuadros con excesivos datos que nadie es capaz de asimilar ni recordar*

dar por mucho tiempo. El sistema informa sobre lo que hay que preocuparse, es decir, sobre los problemas y anormalidades actuales.

2. *Elimina papeles y datos que no tengan ninguna actualidad.*
3. *Los problemas serán solucionados en los lugares mismos en que estos se produzcan.*
4. *Los informes son fácilmente comprensibles puesto que en ningún caso representan muchas cifras.*

El funcionamiento inicial del proyecto Sync tuvo implicancias importantes. Por una parte, y con el propósito de lograr "una dirección eficaz del sector industrial que requiere del respaldo permanente de una corriente apropiada de información", el 30 de abril de 1973 se creó la Dirección de Informática en la CORFO y se designó a su cargo al gerente general de ECOM Raúl Raimundo Beca [19].

Por otra parte, se concluyó que era necesario implementar una red de teleprocesamiento [15]. Para este efecto, en junio de 1973 se estableció un convenio entre ECOM y ENTEL para planificar e implementar un servicio de trasmisión y teleprocesamiento de datos [20]. Así, "un computador distante puede tener acceso a un archivo de datos centralizados, así como también hay la posibilidad de una comunicación de alta velocidad entre computadores. En este último caso puede señalarse, a modo de ejemplo, que un computador instalado en Concepción podría dialogar con otro de mayor capacidad ubicado en la capital".

dad entre computadores. En este último caso

puede señalarse, a modo de ejemplo, que un

computador instalado en Concepción podría

dialogar con otro de mayor capacidad ubicado

en la capital".

DESARROLLO DE PAQUETES

En el período 1971-1973 ECOM transitó desde el desarrollo de proyectos "a la medida" hacia la construcción de paquetes de software genéricos. Por ejemplo, debido a la gran cantidad de sistemas de pagos de sueldos, cuentas corrientes y contabilidad; y la necesidad de ahorrar esfuerzos de programación, análisis y diseño; ECOM decidió desarrollar sistemas genéricos entonces llamados paquetes: "Sistema Generalizado de Pago de Sueldos", "Sistema Normalizado de Cuentas Corrientes Bancarias", "Sistema Contable".

La primera versión del "Sistema Generalizado de Pago de Sueldos" estuvo disponible en septiembre de 1971 y una segunda versión optimizada se liberó en junio de 1973. El modelo básico distinguió como elementos fundamentales los haberes (pagos), los descuentos legales o impositivos y los descuentos varios (compromisos del funcionario). El sistema contempló los archivos Maestro y de Movimiento y los subsistemas de confección y actualización correspondientes

(ACTAM y ACTAMOV). Un tercer subsistema (PROSUELDO) procesaba los archivos y producía el Archivo de Liquidación con los sueldos de los funcionarios. La construcción modular y el uso de tablas permitían adaptar la estructura central con subsistemas optativos para casos particulares [21].

El "Sistema Normalizado de Cuentas Corrientes Bancarias" se desarrolló por iniciativa de la Comisión Bancaria de Procesamiento de Datos, a través de un convenio entre el Banco del Estado y ECOM y la participación de la Superintendencia de Bancos. La motivación era normalizar para facilitar el proceso de fusiones, estandarizar en la banca estatizada y actualizar algunos bancos con déficit y mal estado de las máquinas especializadas de "registro directo". Las funciones del sistema eran: control de cuentas corrientes, actualización diaria, control de retenciones, liquidación a fines de ejercicio, emisión de estados de cuentas y de informes relativos a balances, estadísticas, etc. Respecto del Sistema, ECOM prestaba los siguientes servicios: preparación de datos, operación, análisis y programación, mantenimiento y capacitación para funcionarios de los usuarios [22].

El "Sistema Contable" abordó el problema que deben enfrentar los ejecutivos para la toma de decisiones. La aplicación fue diseñada de manera que la CORFO y sus filiales uniformaran los procedimientos contables. En una primera etapa se desarrolló un Sistema Contable Básico con algunos agregados: presupuesto de caja, cuadros de costo, y algunos indicadores económicos financieros [23]. La segunda etapa, Contabilidad de Gestión, permitiría comparar la producción con su costo detectando los puntos débiles y permitiendo corregir a tiempo. El Sistema se estructuró con los subsistemas de Validación y Cuadratura de documentos contables, Maestro de Cuentas, Diario-Mayor y Balance, Análisis de Cuentas, Lógica y Nombre de Análisis [24] (**Figura 6**).

Otra línea de desarrollo fue la Programación Automática, es decir, la construcción de software que generaba programas de aplicación de uso habitual. Los generadores de programas traducían una especificación en un "lenguaje orienta-

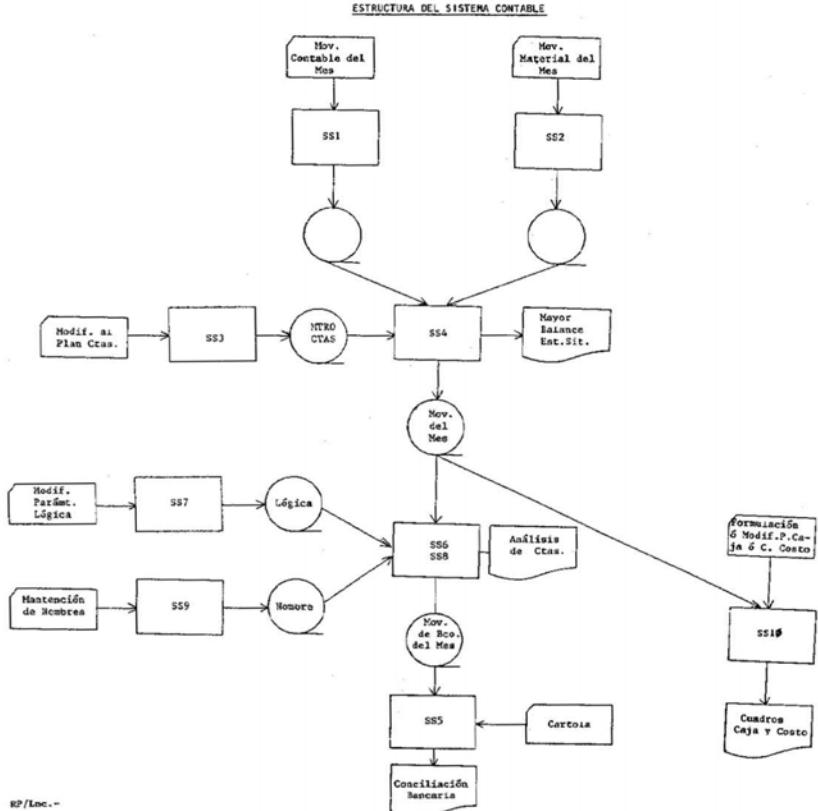


FIGURA 6.
ESTRUCTURA DEL SISTEMA CONTABLE.

do a problema" a instrucciones en un lenguaje de programación tradicional. Al respecto se desarrollaron generadores de programas de validación (GENEVAL) [25], de edición de informes y de analizadores sintácticos. Su generalización se logró automatizando la producción de los programas generadores [26].

DEPARTAMENTO DE INVESTIGACIÓN Y DESARROLLO

Muchos de los proyectos más importantes de ECOM fueron desarrollados o contaron con el apoyo del Departamento de Investigación y Desarrollo (INDE) de la Gerencia de Ingeniería. El Departamento fue fundamental en la investigación aplicada que se necesitó para desarrollar los proyectos y para su divulgación a través de reportes técnicos, boletines y cursos.

Los reportes técnicos se publicaron desde el año 1970 y en ellos se encuentran los fundamentos y los avances del desarrollo de los diferentes proyectos (**Tabla 3**). Los documentos permiten apreciar el trabajo de los investigadores e ingenieros en los distintos proyectos: sueldos, procesamiento de encuestas, bibliotecas, control de proyectos, RUN, recuperación de información, generación automática de programas, teleprocesamiento, etc.

El Boletín del INDE comenzó a publicarse en junio de 1972 con un propósito de divulgación más extensa que los reportes técnicos. Por esa misma razón la extensión de los artículos es más breve y sus contenidos son más generales. Los índices de los primeros seis boletines (**Tabla 4**), permiten apreciar prácticamente a los mismos autores y los mismos temas de los reportes técnicos pero dirigidos a un público más amplio, incluso fuera de ECOM. De hecho, en noviembre de 1972 el boletín alcanzó un tiraje de 900 ejemplares convirtiéndose en un importante medio de información de la comunidad informática.



AUTOR	TÍTULO	FECHA
Avilés, H.	Norma de presentación de informes	72/04
Brokerling, E.	Procesamiento de encuestas mediante computador	70/12
Munita, G.		
Brokerling, E.	Alternativas de Teleprocesamiento. Proyecto Dirección Industrial CORFO	72/07
Calderón, B.	Documentación general del paquete de sueldos	72/03
Brokerling, E.		
Calderón, B.	Estudio preliminar sobre programas generalizados de aplicación estadística	72/05
Duffau, E.	Recuperación de Información	71/05
	Programa graficador de cartas Gantt	72/05
Gamboa, F.	Reglas prácticas para el uso del generador de validación	72/05
Letelier, G.	Programa generador de programas de validación	72/06
Gamboa, F.		
Munita, G.	Documentación de Sistemas. VI: tablas de decisiones	70/06
	Norma sobre preparación de publicaciones	72/04
Pinedo, J.	Sistema generalizado para procesamiento de encuestas por computador, SISTAB	72/07
Segal, S.	PCS, un sistema integrado para el control de proyectos	70/09
	Los métodos de trayectoria crítica y su relación con el proceso administrativo	71/09
	Características de PROMIS y comparación con CPS	72/05
	Ejemplos de PCS	72/05
	Manual de PCS	72/05
	Ejemplo PROMIS	72/06
	Modelo de ubicación de satélites. Unesco. Descripción general del sistema computacional	72/07
Stern, P.	Comparación de datos de alumnos contra un archivo del Registro Civil	71/11
Duffaut, E.		
Vázquez, R.	Resolución de problemas de Bibliotecas por medio de computadores. Sistema BIB1	72/03
	Sistema BIB1 para el control de bibliotecas. Documentación para usuarios	72/03
	Aspectos básicos del control de inventario	72/05

TABLA 3.

REPORTES TÉCNICOS DEPARTAMENTO DE INVESTIGACIÓN Y DESARROLLO.

nacional, razón por la cual se abrió a la publicación de contribuciones provenientes desde fuera de ECOM.

Por otra parte, durante 1971 y 1972 el Departamento realizó diversas charlas y cursos destinados a difundir el empleo de programas de uso general. Particularmente en 1972 se realizaron cursos sobre PCS, MPS, GPSS y GENEVAL, tanto dentro como fuera de ECOM. En 1973 se realizó el programa "Capacitación IRIS-80" [27]. El programa contempló un ciclo de difusión y un ciclo regular de cursos. El ciclo de difusión se realizó a través de seminarios de diez horas orientados fundamentalmente a ejecutivos. El ciclo regular estuvo dirigido a personas que deberían trabajar directamente en los sistemas que serían procesados en el equipo IRIS-80. Al respecto, se ofreció

un programa por tipo de especialistas: jefes de servicios (2.5 días), analista (20.5 días), programador (33 días) y programador y analista avanzado (63 días).

RELACIÓN CON UNIVERSIDADES

Durante el período 1971-1973 ECOM se relacionó permanentemente con las universidades. En el caso de la Universidad de Chile esa relación se manifestó principalmente en docencia y en memorias de título. Varios ingenieros e investigadores de ECOM fueron profesores de cursos de Computación en la Escuela de Ingeniería. En

particular Hugo Segovia fue profesor de "Sistemas de Información", Pablo Fritis de "Estructuras y Procesos de la Información", René Barros y Víctor Canales de "Introducción a la Computación" y Alberto Ulriksen y Hernán Avilés de "Simulación".

ECOM fue también el lugar donde varios estudiantes de la Universidad de Chile trabajaron y realizaron sus memorias en el período 1971-1973 aunque sus titulaciones fueron posteriores (**Tabla 5**). Las memorias estuvieron asociadas a los proyectos Syncro, Censo, Generación Automática de Programas, Inventarios, etc. Y cubrieron distintas carreras de Ingeniería: Civil, Industrial, Matemática, y de Ejecución en Procesamiento de la Información (la primera carrera de ingeniería en el área en Chile).

Nº	FECHA	AUTOR	TÍTULO ARTÍCULO
1	26/06/72	Calderón, Benjamín	Sistema generalizado de pago de sueldos
		Letelier, Gabriel	GENEVAL
2	26/07/72	Munita, Germán	Ánálisis histórico del rendimiento de equipos computacionales
		Letelier, Gabriel	Estado actual de GENEVAL
3	28/08/72	Avilés, Hernán	Sistemas, modelos y simulación
		Duffau, Etienne	Corrección de claves numéricas mediante el uso de dos dígitos verificadores
		Cabrera, Carlos	Sistema normalizado de cuentas corrientes
		Calderón, Benjamín	Programas relacionados al paquete de sueldos
		Letelier, Gabriel	Datos estadísticos acerca de la utilización del generador de programas
		Gamboa, Fernando	de validación
		Guzmán, Felipe	Proposición de sistema de información para mantención de equipos
		Benadof, Isaquino	¿Qué es la Cibernética?
		Avilés, Hernán	Dinámica de sistemas
4	26/09/72	Duffau, Etienne	Versión preliminar de un soundex español
		Calderón, Benjamín	Sistema de identificación del paquete de sueldos
		Letelier, Gabriel	Evaluación de un programa generado por GENEVAL mediante "PPE"
		Benadof, Isaquino	Herramientas y cibernética
		Cárdenas, Hernán	Modelo de simulación de estructura de archivos
		Munita, Germán	Las economías de escala y la ley de Grosch
5	26/10/72	Brain, Hernán	Conceptos generales del control de inventarios
		Stern, Luis	PROMIS, un sistema de control de proyectos
		Calderón, Benjamín	Sistema básico de codificación del paquete de sueldos
		Letelier, Gabriel	Aplicabilidad de META-POL a la generación de programas
		Benadof, Isaquino	¿Qué es la ergonomía?
		Munita, Germán	Los nuevos computadores de ECOM
6	24/11/72	Duffau, Etienne	Sistemas de Identificación I
		Segal, Sergio	Formulación y solución computacional del problema de la dieta
		Cabrera, Carlos	Sistema normalizado de cuentas corrientes
		Calderón, Benjamín	Registros de entrada al paquete de sueldos
		Letelier, Gabriel	Características de una 3ra. Versión de GENEVAL

TABLA 4.

ÍNDICES BOLETÍN DEL DEPARTAMENTO DE INVESTIGACIÓN Y DESARROLLO.

AÑO	NOMBRE	TÍTULO	ESPECIALIDAD
73	Avilés, Hernán	Dinámica de sistemas una técnica de simulación	Matemática
73	Stern, Luis	Estudio, prueba e implementación de un sistema de control de proyectos con nivelación de recursos	Industrial
74	Cárdenas, Hernán	Evaluación de técnicas de organización de archivos	Industrial
74	Gamboa, Fernando	Programación automática de generadores de programas y analizadores sintácticos	Procesamiento Información
74	Castro, Eduardo Ibáñez, Pablo	Algoritmos de corrección de errores aplicados al procesamiento de personas del XIV Censo	Procesamiento Información
75	Brain, Hernán	Un sistema computacional generalizado de control físico y valorizado de inventarios	Industrial
75	Letelier, Gabriel	Diseño y construcción de un generador de programas de validación	Civil

TABLA 5.

MEMORIAS REALIZADAS EN ECOM POR ESTUDIANTES DE LA UNIVERSIDAD DE CHILE.

La relación con las universidades se produjo también en el marco de la Asociación Chilena de Centros Universitarios de Computación (ACUC), creada en diciembre de 1970, donde ECOM participó como miembro invitado con derecho a voz. De hecho, en el primer boletín de la ACUC, de julio de 1973, Germán Munita escribió un artículo respecto de los computadores disponibles en ECOM y de los futuros computadores IRIS [6].

SUBGERENCIA EN CONCEPCIÓN

En 1971 CORFO Biobío aprobó la idea de crear una instalación informática regional para apoyar

las empresas del Área Social, algunas de las cuales ya poseían computadores y personal capacitado [28]. ECOM acogió la iniciativa y creó una subgerencia que dependía del Gerente General, y contó con el apoyo de la Gerencia de Administración y Finanzas. Posteriormente, en 1972, ECOM aprobó el proyecto "Centro de Informática Lautaro" que albergaría un IRIS-60 y minicomputadores MITRA-15 para teleproceso. El proyecto contempló un edificio de mil metros cuadrados con empleo de acero, vidrio y cemento y con un mural de Julio Escámez en el hall de acceso.

La subgerencia estuvo a cargo del ingeniero Óscar Sáez y se contrató una docena de funcionarios, la mayoría ingenieros y técnicos (entre ellos siete analistas y un programador) y fueron

capacitados en las herramientas de los IRIS y MITRA. Dos de ellos fueron enviados a capacitarse a Francia.

La subgerencia realizó proyectos en el marco de convenios con la siderúrgica CAP, la minera ENACAR y la textil Bellavista Tomé. En este último caso, en 1973 ECOM propuso un proyecto de informatización integral de la fábrica textil que fue aprobado por la Asamblea de Trabajadores de la empresa. El proyecto incluía tanto funciones administrativas como de apoyo a la producción.

CONCLUSIONES

LA REVISIÓN DE LA ESTRUCTURA, LOS RECURSOS Y LOS PROYECTOS DE LA EMPRESA NACIONAL DE COMPUTACIÓN E INFORMÁTICA (ECOM) DURANTE EL PERÍODO 1971-1973, PERMITE DISTINGUIR ELEMENTOS DE CONTINUIDAD Y CAMBIO CON RESPECTO AL PERÍODO INICIAL 1968-1970 DE LA EMPRESA DE SERVICIO DE COMPUTACIÓN (EMCO).

ECOM CONTINUÓ CUMPLIENDO LOS PROPÓSITOS FUNDACIONALES DE PROPORCIONAR SERVICIOS COMPUTACIONALES A LAS INSTITUCIONALES ESTATALES Y CAPACITAR PERSONAL. EN ESTAS DOS DIMENSIONES ECOM RE-PRESENTÓ UN SALTO CUANTITATIVO, AUMENTANDO, TANTO LA CANTIDAD DE COMPUTADORES Y USUARIOS, COMO EL NÚMERO DE PERSONAS ENTRENADAS EN PROGRAMACIÓN, ANÁLISIS E INTERLOCUCIÓN.

POR OTRA PARTE, EL QUEHACER DE ECOM SIGNIFICÓ UN SALTO CUALITATIVO RESPECTO DE LO REALIZADO POR EMCO. EN EFECTO, ENTRE LOS AÑOS 1971 Y 1973, ECOM TRANSITÓ DESDE LA CONSTRUCCIÓN DE SOFTWARE "A LA MEDIDA", HACIA EL DESARROLLO DE SISTEMAS GENÉRICOS (SUELDOS, CUENTAS CORRIENTES, CONTABILIDAD, GENERADORES DE PROGRAMAS) Y DE PROYECTOS DE IMPACTO NACIONAL EN CONJUNTO CON OTRAS INSTITUCIONES ESTATALES, POR EJEMPLO EL CENSO CON EL INE, EL RUN-RNP CON EL REGISTRO CIVIL Y EL MINISTERIO DE JUSTICIA, EL EMBLEMÁTICO PROYECTO SYNC CON LA CORFO E INTEC, Y LOS PROYECTOS CON LAS INDUSTRIAS DE LA REGIÓN DEL BIOBÍO.

EL SALTO CUALITATIVO SE SUSTENTÓ EN EL TRABAJO DE LAS GERENCIAS DE INGENIERÍA Y DE PLANIFICACIÓN Y DESARROLLO Y DEL DEPARTAMENTO DE INVESTIGACIÓN Y DESARROLLO. ALLÍ SE REALIZARON LOS ESTUDIOS, LA INVESTIGACIÓN APLICADA Y LAS PRUEBAS DE LOS PROTOTIPOS DE LOS SISTEMAS DE SOFTWARE. SUS RESULTADOS FUERON DIFUNDIDOS EN EL MEDIO NACIONAL A TRAVÉS DE REPORTES TÉCNICOS, BOLETINES Y CURSOS. ESTE QUEHACER SE RELACIONÓ TAMBIÉN CON LAS UNIVERSIDADES A TRAVÉS DE MEMORIAS, DOCENCIA Y COORDINACIÓN CON SUS CENTROS DE COMPUTACIÓN.

EN SÍNTESIS, EL QUEHACER DE ECOM EN EL PERÍODO COMPRENDIDO ENTRE DICIEMBRE DE 1970 Y SEPTIEMBRE DE 1973 DEJÓ UN VALIOSO LEGADO AL MEDIO COMPUTACIONAL Y TECNOLÓGICO NACIONAL. LA MAYORÍA DE LOS PROYECTOS Y SISTEMAS CONTINUARON VIGENTES, OTROS QUEDARON PENDIENTES Y FUERON REALIZADOS POSTERIORMENTE (NUEVO EDIFICIO, COMPUTADOR PARA LA UTE), Y UNOS POCOS FUERON CANCELADOS (LA COMPRA DE LOS COMPUTADORES FRANCESES, EL PROYECTO SYNC). POR ÚLTIMO, Y QUIZÁS LO MÁS IMPORTANTE, ECOM LEGÓ AL PAÍS UN CONJUNTO DE PERSONAS DOTADAS DE UNA EXPERIENCIA, UNA MÍSTICA DE TRABAJO Y UN ESPÍRITU DE SERVICIO PÚBLICO QUE CONTRIBUYERON AL DESARROLLO Y PROGRESO DE LA DISCIPLINA INFORMÁTICA EN CHILE Y EN ALGUNOS PAÍSES DEL EXTRANJERO.



El 11 en ECOM: La experiencia de uno de sus trabajadores

NN comenzó a trabajar en la Empresa Nacional de Computación (ECOM) en el año 1971. Siendo aún estudiante de Ingeniería de la Universidad de Chile, su situación económica lo obligó a dejar sus estudios y aprovechó una oportunidad que se presentó para trabajar como operador de equipos periféricos del computador. Su primer día de trabajo consistió en vigilar el normal funcionamiento de una impresora 1403 de uno de los computadores IBM/360, evitando los atascos y alimentándola de papel que entonces se elaboraba en hojas con prepicio y en formularios continuos.

Por su buen desempeño y dedicación al trabajo asumió prontamente responsabilidades mayores en la operación de los computadores de la empresa y le correspondió supervisar sistemas computacionales que apoyaban la administración de las empresas del Estado y que se procesaban en alguno de los tres computadores IBM (uno de los cuales residía en Endesa). Eran los años del gobierno del Presidente Salvador Allende y NN contribuía con su trabajo a apoyar entusiastamente la construcción del socialismo en Chile.

La empresa, tratando de diversificar la procedencia de sus equipos y evitar la dependencia de un único fabricante, adquirió un computador Burroughs 3500 y encargó dos computadores franceses CII (Iris-80 e Iris-60). La operación de los equipos de entonces era compleja y muy delicada por lo que se requería actualización y entrenamiento permanente. El día 11 de septiembre de 1973, NN, que vivía en la comuna de San Miguel, acudió muy temprano a un curso de capacitación que se realizaba en el edificio "Sol de Chile" en la calle Huérfanos. Se extrañó que sus compañeros no llegaran, hasta que alguien le informó que se estaba produciendo un golpe de Estado.

Estando muy cerca de La Moneda, instintivamente se dirigió hacia allá y observó que estaba siendo rodeada de tanquetas. Desde una de ellas se

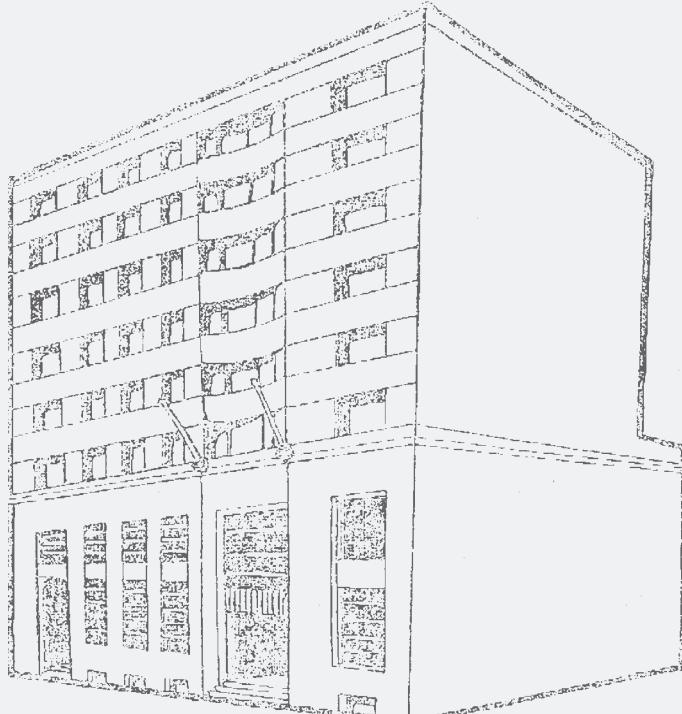
disparó una ráfaga por lo que decidió regresar al edificio principal de ECOM. Las horas fueron mostrando la brutalidad del golpe y desde las oficinas observó a militares sacando y apresando trabajadores en los edificios cercanos, especialmente en el Ministerio del Trabajo ubicado en la esquina de las calles Huérfanos y Morandé.

NN, junto a una veintena de funcionarios, la mayoría (pero no todos) simpatizantes del gobierno de Allende, decidió quedarse y tratar de proteger y defender las instalaciones. El toque de queda duró varios días y la permanencia se hizo peligrosa y difícil. Cerraron las ventanas y durante la noche apagaban las luces para no despertar la atención de las patrullas militares. Para alimentarse tuvieron que aprovisionarse de tallarines desde un casino que se encontraba en el último piso del edificio y los prepararon en una cocinilla. En una ocasión fueron sorpresivamente interrumpidos por el sonido de un mensaje que llegó a un terminal de télex, exigiéndoles que reportaran la cantidad y la lista de personas que permanecían en la empresa.

Decidieron no responder. La espera se hizo infructuosa y se retiraron sigilosamente al levantarse el toque de queda.

El regreso al trabajo fue muy tenso. NN decidió presentarse, a pesar de que eran conocidas sus simpatías por el Gobierno de Allende. Algunas personas no regresaron y otras fueron despedidas. Esperando lo peor, NN fue gratamente sorprendido por el encargado del equipo Burroughs 3500 que, conociendo su experiencia técnica, le ofreció trabajar en la operación de ese computador. NN aceptó la propuesta y durante un par de años tuvo un desempeño que fue destacado y reconocido. De hecho, llegó a estar a cargo de dos computadores 3500, de los más grandes y sofisticados de la época.

Lamentablemente, esta historia no tuvo un final feliz. Uno de los computadores 3500 fue asignado a una de las ramas de las fuerzas armadas, que exigió la expulsión de NN. En una reunión surrealista, el Gerente no encontraba la justificación para despedir a un funcionario que tenía una impecable y muy destacada trayectoria laboral. ■



AGRADECIMIENTOS Y DEDICATORIA

Al personal de las Bibliotecas de las Escuelas de Ingeniería y de Economía de la Universidad de Chile, de CORFO, del Congreso Nacional y de la Biblioteca Nacional por facilitar el acceso a las fuentes escritas. Gracias también a Germán Munita y Óscar Sáez por la valiosa información acerca del Departamento de Investigación y Desarrollo y de la Subgerencia del Biobío. Finalmente, un especial reconocimiento a mi profesor Hugo Segovia (Q.E.P.D.) por su disertación en el año 2008 (en un acto de conmemoración de los cuarenta años de la creación de ECOM) y por una entrevista en 2009. A su memoria dedicamos este artículo, personalizando en él, el gran aporte que realizó al país ECOM y sus trabajadores entre los años 1971 y 1973.

REFERENCIAS

- [1] Álvarez, Juan. "Empresa Nacional de Computación: antecedentes, creación y primeros años". Revista Bits de Ciencia N° 9, Primer Semestre 2014.
- [2] Marfán, Álvaro; Friedmann, Efraín; Sáenz, Ricardo. "Experiencia en el sector público de Chile sobre el procesamiento electrónico de datos con computadores electrónicos". Actas I Conferencia Latinoamericana de autoridades gubernamentales de Informática". Buenos Aires, abril 1970.
- [3] Diario La Nación. "ECOM: nuevo nombre para una empresa que surge", 16 de enero de 1971.
- [4] CORFO. "Información presupuestaria empresas e institutos filiales CORFO Año 1972". Julio 1971.
- [5] Presidencia de la República. "Mensaje del Presidente Allende ante el Congreso Pleno". 21 de mayo de 1972.
- [6] ACUC. "Revista de la Asociación Chilena de Centros Universitarios de Computación". N°1, julio 1973.
- [7] ECOM. "Características principales de los sistemas IRIS-80 e IRIS-60". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [8] EMCO. "EMCO Capacitación". 1970.
- [9] Segovia, Hugo. "Gerencia de Ingeniería, Gerencia de Investigación y Desarrollo 1971-1973". Presentación en acto "A 40 años de la creación de la Empresa Nacional de Computación". DCC, U. de Chile, diciembre 2008.
- [10] Diario La Nación. "El Censo, a las computadoras". 23 de abril de 1970.
- [11] Instituto Nacional de Estadísticas. "XIV Censo Nacional de Población y II de Vivienda 1970". [http://www.ine.cl/canales/usuarios/cedoc_online/censos/pdf/censo_1970.pdf](http://www.ine.cl/canales/usuarios/cedoc_oline/censos/pdf/censo_1970.pdf)
- [12] OCOM; ECOM. "Proyecto Rol Único Nacional y Registro Nacional de Población". Agosto 1972.
- [13] OCOM; ECOM; Subsecretaría de Justicia. "Antecedentes para la creación de un registro automatizado de población en Chile". Actas Seminario "Sistemas de Información Administrativos", U. de Chile. Julio 1971.
- [14] Stern, Pinchas; Duffau, Etienne. "Un proyecto piloto para la creación de un archivo de personas". Actas I Congreso Iberoamericano de Informática". Buenos Aires, mayo – junio 1972.
- [15] CORFO. "Proyecto Syncro. Conceptos y Práctica del Control; una experiencia concreta: la Dirección Industrial en Chile". Febrero 1973.
- [16] Presidencia de la República. "Mensaje del Presidente Allende ante el Congreso Pleno". Páginas 412-415. 21 de mayo de 1973.
- [17] Medina, Eden. "Revolucionarios cibernéticos". Editorial LOM. 2013.
- [18] ECOM. "Proyecto Syncro – Sistema Cyberstride". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [19] Diario El Mercurio. "Cinco nuevas reparticiones creó Vuskovic en la CORFO". 30 de abril de 1973.
- [20] Diario El Mercurio. "Convenio ENTEL-ECOM: Planifica Servicio de Transmisión y Teleprocesamiento de Datos". 27 de junio de 1973.
- [21] ECOM. "Sistema generalizado de pago de sueldos". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [22] ECOM. "Sistema normalizado de cuentas corrientes bancarias". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [23] ECOM. "Sistema Contable Computacional". 12 de junio de 1972.
- [24] ECOM. "Sistema contable". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [25] Letelier, Gabriel. "Diseño y Construcción de un Generador de Programas de Validación". Memoria para optar al título de Ingeniero Civil de la Universidad de Chile. 1975.
- [26] Gamboa, Fernando. "Programación Automática de Generadores de Programas y Analizadores Sintácticos". Informe para optar al título de Ingeniero de Ejecución en Procesamiento de la Información de la Universidad de Chile. 1974.
- [27] ECOM. "Programa capacitación IRIS-80". Actas Conferencia "Facilidades de procesamiento electrónico de datos para empresas y organismo gubernamentales". Junio 1973.
- [28] Sáez, Óscar. Comunicación personal. Junio 2015.

¿EXISTE UNA INDUSTRIA TI EN CHILE?

"No me parece mal ser maestro chasquilla, desatornillador en mano andar arreglando entuertos con mucha imaginación. De hecho esa es una de mis formas de ganarme la vida. La pregunta es si podremos desarrollar este país hacia el futuro, con competitividad internacional, en base a pura artesanía. Ojalá que cuando la tecnología nacional se haga necesaria al borde de una catástrofe, podamos encontrarla e n algún lado".

José Miguel Piquer, Columna en Revista Informática, 1996.





JOSÉ MIGUEL PIQUER

Profesor Asociado Departamento de Ciencias de la Computación, Universidad de Chile. Director Técnico de NIC Chile y socio fundador de la consultora IT-Talk Ltda. y de E-voting Chile SpA, dedicada a servicios de votación electrónica. Entre 2011 y 2013, fue el encargado del desarrollo de proyectos científicos de transferencia en INRIA Chile, centro de excelencia internacional, donde se busca establecer un puente entre la investigación y el desarrollo productivo.

jpiquer@it-talk.cl

Por más de veinte años he albergado la esperanza que la industria TI chilena un día se transforme en motor del desarrollo y posicione a Chile en la lista de los países líderes en generación de software en el mundo. A comienzos de los noventa, mi generación de profesores estaba recién llegando de sus doctorados a instalarse en la naciente democracia chilena, y soñamos que podíamos cooperar con la industria para lograr este liderazgo [1]. Fue obviamente una ilusión pasajera y un sueño muy inocente, pero siempre he creído que, en el largo plazo, es posible: nuestras universidades forman profesionales de nivel mundial, nuestros departamentos de computación son de los mejores en América Latina y están en una buena posición en el mundo. Además, llevan haciendo esto ya por muchos años, creando varias generaciones de profesionales. Entre tanto, el país se ha desarrollado enormemente, y tenemos varias industrias que hoy compiten por los primeros lugares en el mundo.

Es decir, debiera estar todo lo necesario para el éxito: una oferta de profesionales y de investigación entregada por buenas universidades, y una demanda de alto nivel y complejidad basada en industrias líderes a nivel mundial. Es el plan que estuvo en la base de la creación de los *cluster* propuestos por el Consejo Nacional para la Innovación en 2005, y que buscaba impulsar la innovación y el emprendimiento para cambiar el modelo de desarrollo del país, pasando de las materias primas a productos con valor agregado. Aunque las TI nunca fueron un *cluster*, justamente porque nuestra industria no es una de las que compite a nivel mundial, siempre estuvieron en la lista de lo que llamaron "tecnologías habilitantes", cuya demanda debía aumentar mucho al desarrollar los verdaderos *cluster*.

Al mirar el escenario actual, diez años de esfuerzos en innovación en Chile no parecen haber dado los frutos esperados. Seguimos siendo el mismo país esclavo de los recursos naturales y sus precios, como lo demuestra la desaceleración económica que sufrimos desde 2014, simplemente debido a una baja sostenida en el valor internacional de los *commodities*. Esto se refleja igualmente en la industria TI chilena: nuestros grandes proveedores son muy buenos integradores de sistemas, que hoy funcionan como socios estratégicos de las empresas al mantener sus sistemas operando, pero casi no existen ejemplos de servicios o desarrollos de software innovadores que compitan a nivel mundial. ¿Tenemos alguna esperanza aún?

Algunas luces quedan en el sistema, que parecen indicar que Chile no ha perdido todas sus oportunidades. Veamos las razones que nos dan esperanzas:

- 1) La exportación de servicios TI ha crecido en los últimos años, posicionando a Chile como punto de destino de inversiones interesantes [2], así también como un país plataforma de *nearshoring* con varias transnacionales instalando sus centros de desarrollo TI e incluso algunos centros de I+D en Chile. Aunque los números todavía son pequeños (100 millones de dólares en 2010) parece una tendencia al alza importante. Esta tendencia, de producir software para el mundo, es muy positiva, dado que uno de los grandes frenos a la industria local siempre ha sido la falta de sofisticación de los clientes locales. No es posible desarrollar software de clase mundial para un cliente que quiere que le amarre con alambritos un sistema a la rápida (y eso es lo que está dispuesto a pagar).

2) La demanda por aplicaciones móviles volvió a generar un mercado emergente, parecido a lo que era la demanda por aplicaciones web en los noventa. En esos mercados siempre nos hemos movido bien, porque requieren productos rápidos y sucios, pero que funcionen bien. Esta vez no fue la excepción (y creo que *Startup Chile* ayudó mucho a esto). El tema es que Chile aparece en las listas de países donde es bueno mandar a hacer aplicaciones móviles, y hemos ganado cierta fama en ese nicho. Lo difícil es mantener ese empuje en el tiempo, y saber ir migrando hacia sistemas más maduros y estables, donde la improvisación y la suciedad ya no son aceptables.

3) He visto a varios de nuestros exalumnos involucrados en emprendimientos cada vez más sofisticados, innovadores y riesgosos. En los viejos tiempos fueron Tecnonautica (comprado por Telefónica), Mapcity, Acepta.com y Atentus. Hoy existen muchos más como esos, en juegos para móviles, medios de pago, y varios otros. De todo este ecosistema de gente innovadora, algo muy bueno terminará saliendo.

Tal vez era necesario que pasara el tiempo, que los grupos se consolidaran, que el país madurara. Pero estamos de vuelta en lo mismo: una temprana infancia de una industria emergente que no parece lograr consolidarse nunca. O tal vez uno de estos días...

Ojalá que, de todos estos proyectos, salga ese ejemplo de éxito que necesitamos, que demuestre que se puede desarrollar tecnología de nivel mundial en este perdido lugar al fin del mundo (pero con buena conexión a Internet!).

En todo este sueño, el Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile hizo sus aportes: formamos muy tempranamente profesionales en tecnologías innovadoras e importantes, ayudamos en las conexiones a Internet del país y hemos difundido las tecnologías relevantes cada vez que podemos. Sin embargo, me parece que podríamos hacer mucho más, incentivando a nuestros estudiantes a emprender, apoyando el despliegue de nuevas tecnologías en el país, y acompañando a la industria en el difícil camino de la innovación tecnológica.

Hoy el DCC no se ve muy presente en los esfuerzos de transferencia que la misma Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile ha comenzado, como el área de innovación del Proyecto 2030 o proyectos concretos internos como el Fab Lab. En algún momento tuvimos roces con la Facultad porque nuestros académicos hacían muchos proyectos aplicados y poca investigación. Parece que ahora se invirtieron los roles, tal vez se nos pasó la mano en insistir tanto en las publicaciones internacionales, el asunto es que nadie parece tener el tiempo que estos esfuerzos requieren.

Cooperar con las empresas y con los emprendedores no es fácil ni eficiente. El entorno chileno es todavía adverso a la innovación y al riesgo, nadie cree que realmente hay valor en desarrollar tecnología y arriesgarse en áreas inexploradas. Por lo tanto, no existe ningún incentivo concreto para ir en esa dirección, siendo lo normal hoy dormirse en los laureles y revisar cuál es el próximo concurso CONICYT donde postular. Si no rompemos esa rutina, demoraremos veinte años más en tener una verdadera industria TI. ■



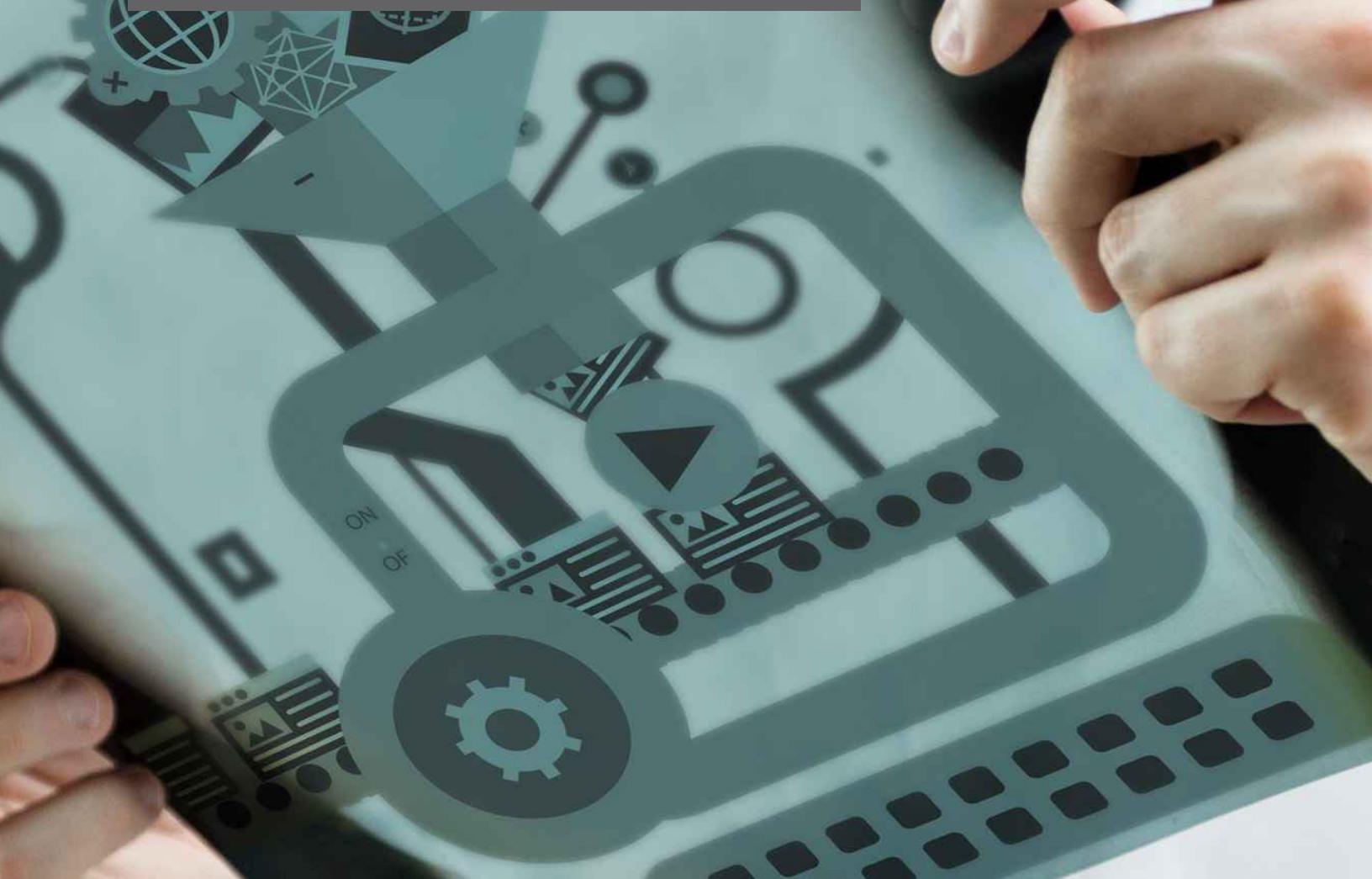
REFERENCIAS

[1] Baeza-Yates, R. A., Fuller, D. A., Pino, J. A., Goodman, S. E. (1995). Computing in Chile: the jaguar of the Pacific Rim?. *Communications of the ACM*, 38(9), 23-28.

[2] R. A. Hernández, N. Mulder, K. Fernandez-Stark, P. Sauvé, D. L. Giral, F. M. Navia (Eds.), *Latin America's emergence in global services* (pp. 177–195). Santiago: UN, 2014.

RADIOGRAFÍA A LA INDUSTRIA DEL SOFTWARE EN CHILE Y SUS DESAFÍOS

El mercado tecnológico chileno es uno de los más desarrollados de Latinoamérica, representando un 1,7% del PIB, según informes recientes de IDC. No obstante, su potencial de crecimiento es aún extenso si consideramos que en los países desarrollados, el porcentaje invertido en tecnología llega al 2,5% del PIB.





MARÍA EUGENIA RIQUELME

Ingeniero en Comercio Internacional, con Diplomado en Gestión y Desarrollo de Negocios. Cuenta con 18 años de experiencia en gestión de Asociaciones Gremiales, actualmente dirige la Asociación de Empresas de Software y Servicios, GECHS (9 años), siendo responsable de las relaciones institucionales de la Asociación con el sector gubernamental, privado, académico y medios de comunicación, a efecto de contribuir a una mejor imagen, posicionamiento y concreción de los objetivos del sector de las Tecnologías de la Información. Interactúa con las empresas socias y representa sus intereses. Participa en actividades a nivel nacional e internacional para generar redes. Posee amplio conocimiento de la industria, lo que le ha permitido gestionar diversos proyectos asociativos para empresas tecnológicas. Ha preparado y participado en diversos eventos de networking y meetup, con invitados nacionales e internacionales.

meriquelme@gechs.cl

En Chile, la industria del software está formada por PyMEs que en general realizan venta de productos tecnológicos e integración, pero no alta tecnología, que es lo que en Gechs como Asociación estamos buscando cambiar. Históricamente hemos sido un gremio de servicios, pero en el último año, gracias a un proyecto ejecutado junto a Corfo, denominado SaaS Generation, estamos buscando pasar de ser desarrolladores a medida a ser desarrolladores de "productos" tecnológicos que se puedan vender como servicios escalables a nivel global.

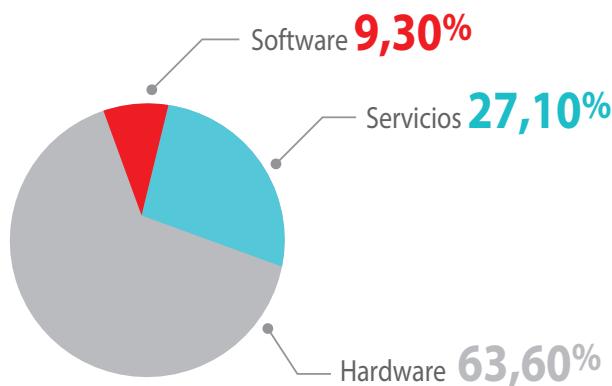


FIGURA 1.
COMPOSICIÓN DE LA INDUSTRIA TECNOLÓGICA CHILENA.

ESTRUCTURA DE LA INDUSTRIA TIC

En Chile, de las aproximadamente 950 mil empresas registradas en el Servicio de Impuestos Internos, hay 4.700 con giro TIC, de las cuales hay desde microempresas que se dedican a vender insumos para equipos, hasta empresas más formales que desarrollan software.

De esas 4.700, estimamos que deben haber unas 1.600 empresas que realizan servicios más sofisticados, tanto servicios tecnológicos, como desarrollo a medida o empresas de base tecnológica, que son las que pueden crecer exponencialmente si logramos abrirlas a nuevos mercados. El modelo de Software as a Service (SaaS), que estamos promoviendo entre los miembros de la Asociación, así como en jóvenes innovadores y emprendedores, va en esa línea.

Según datos de IDC, de las empresas TIC en Chile, un 95% de ellas son nacionales y un 5% son de origen internacional, principalmente en el segmento de mayor tamaño.

Al desagregar la Industria tecnológica chilena, podemos ver que está compuesta por los sub-sectores: Software con un 9,3% de participación, Servicios con un 27,1% y Hardware, el más relevante, con un 63,6% (**Figura 1**).

Para contar con una industria tecnológica mejor mapeada y con datos estadísticos certeros –pues hoy ninguna entidad los genera de manera sistemática– es que comenzamos a trabajar en un proyecto estadístico con País Digital, fundación sin fines de lucro que tiene como finalidad la

investigación, difusión, promoción y desarrollo de los distintos aspectos relacionados con las ciencias tecnológicas.

NUESTRO DESAFÍO

En Gechs aspiramos a que Chile sea la California de Sudamérica, con servicios más sofisticados.

Económicamente contamos con empresas extractivas que son agregadoras de valor, pero tenemos que generar una industria creadora de valor, para atraer a profesionales TIC que sean capaces de cambiar nuestra realidad.

En esta línea, buscamos hacer crecer a las empresas de tecnología para que mediante la apertura de mercados internos y externos facturen más; para lo cual contamos con una mesa de negociación con el Gobierno y otra de internacionalización, en la cual estamos generando distintas misiones a otros países, para mostrar lo que hacen nuestras empresas y empaparnos de experiencias internacionales.

Hemos dicho que la industria de las tecnologías digitales y sobre todo el software tiene un gran potencial, que todavía nos falta por desarrollar. Todos tenemos claro, partiendo por el Ministerio de Economía, la importancia del uso de las tecnologías de la información como motor de desarrollo de las economías modernas y es por eso que actualmente está en curso el Programa Estratégico de Gobierno para la Industria Nacional de TIC y Servicios Globales que busca generar capacidades de empaquetamiento y escalamiento, de manera de convertir las soluciones particulares actuales en productos que puedan ser comercializados a nivel mundial.

El Programa SaaS Generation, cofinanciado por Corfo, se desarrollará durante todo el 2015 y consiste en buscar nuevas oportunidades de crecimiento de nuestra industria, a través del desarrollo de capacidades en modelos de negocio diferentes a los que caracteriza a la industria chilena del software (basada en la tecnología a medida). De esta manera, las empresas participantes, podrán capturar mejor el valor de su trabajo, más allá del mercado nacional, de modo de convertir a Chile en un líder en el desarrollo tecnológico. Todo ello a través de la adquisición de capacidades en el desarrollo de Software as a Service para crear negocios escalables a nivel internacional.

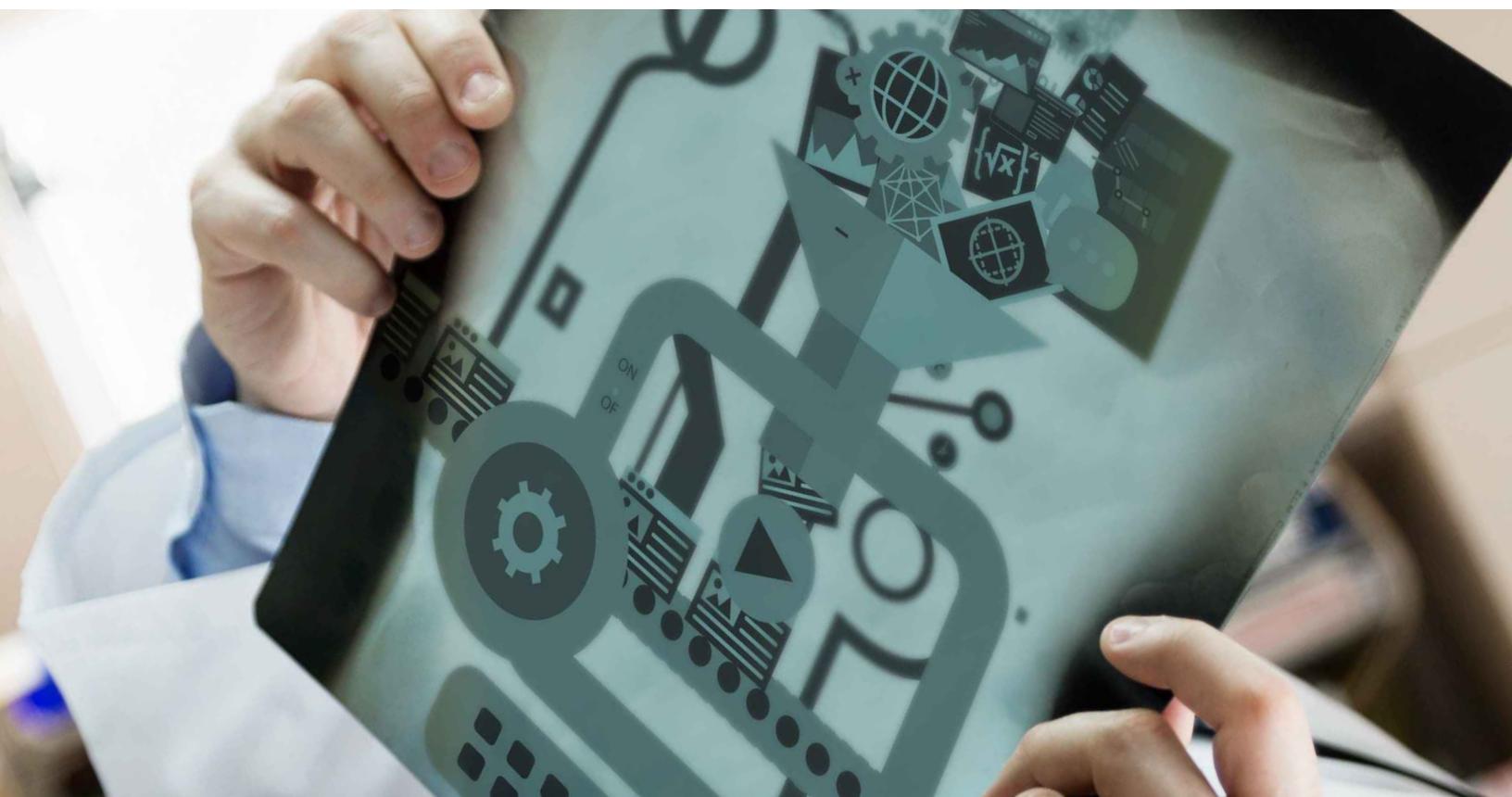
En cuanto a otros desafíos que nos restan por vencer, dos de los principales son descentralizar la industria TIC y aumentar el porcentaje de participación femenina.

La industria TIC se encuentra altamente concentrada en la Región Metropolitana con un 87% de las empresas, seguida por la Región de Valparaíso

con un 8% y la Región de Concepción con un 4,7% de las empresas. Sin duda este es un aspecto que debiéramos mejorar como país, poniendo más énfasis en generar apoyos para emprendimientos e innovaciones que destaquen en regiones, dado que se han visto ejemplos muy buenos de proyectos tecnológicos y aplicaciones (por ejemplo de Smart Cities) provenientes de esas regiones.

En lo referido a la participación del mundo femenino en la industria TIC, durante 2014, esta participación no alcanzó el 5%. Por ello resulta fundamental identificar y visibilizar las barreras culturales, que no están permitiendo que las mujeres se motiven a formar parte de esta industria, que se caracteriza por ser meritocrática.

Esperemos que este año la participación femenina crezca a tasas muchísimo mayores, tanto por la diversidad de opciones laborales que hay en este mundo tecnológico como por la fuerte necesidad de contar con las habilidades blandas que otorgan las mujeres a las distintas áreas laborales. ■



LA INDUSTRIA DEL SOFTWARE EN CHILE: DESAFÍOS Y OPORTUNIDADES PARA GENERAR UN IMPACTO PAÍS



MARÍA CECILIA BASTARRICA

Profesora Asociada Departamento de Ciencias de la Computación, Universidad de Chile. PhD Computer Science and Engineering, University of Connecticut; Magíster en Ciencias de la Ingeniería, Universidad Católica de Chile; Ingeniera en Informática, Universidad Católica del Uruguay. Líneas de investigación: Ingeniería de software - arquitectura de software, líneas de productos de software, mejora de procesos de software, desarrollo de software dirigido por modelos.

cecilia@dcc.uchile.cl



MAIRA MARQUES

Candidata a Doctora en Ciencias mención Computación y Magíster en Ciencias mención Computación, Universidad de Chile; Ingeniera Química de la Universidad Federal de São Carlos, Brasil. Líneas de investigación: Ingeniería de Software - educación en ingeniería de software, procesos de software y trabajos colaborativos.

mmarques@dcc.uchile.cl



SERGIO OCHOA

Profesor Asociado Departamento de Ciencias de la Computación, Universidad de Chile. Doctor en Ciencias de la Computación, Pontificia Universidad Católica de Chile; Ingeniero de Sistemas, UNICEN, Argentina. Líneas de investigación: Sistemas Colaborativos, Ingeniería de Software -arquitectura de software, mejora de procesos en micro y pequeñas empresas de software, ingeniería de requisitos, educación apoyada con tecnología.

sochoa@dcc.uchile.cl



DANIEL PEROVICH

Profesor Adjunto de Jornada Parcial Departamento de Ciencias de la Computación, Universidad de Chile. Doctor en Ciencias mención Computación, Universidad de Chile. Magíster en Informática e Ingeniero en Computación, Universidad de la República, Uruguay. Líneas de investigación: Ingeniería de Software, Arquitectura de Software, Procesos de Desarrollo y Desarrollo Dirigido por Modelos.

dperovic@dcc.uchile.cl



ALCIDES QUISPE

Profesor Asistente Departamento de Ingeniería Informática, Universidad de Santiago de Chile. Doctor en Ciencias mención Computación, Universidad de Chile; Ingeniero de Sistemas, Universidad Católica de Santa María, Perú. Líneas de investigación: Ingeniería de Software - ingeniería de requisitos, procesos de software, e investigación empírica en ingeniería de software.
alcides.quispe@usach.cl



ROMAIN ROBBES

Profesor Asociado, Departamento de Ciencias de la Computación, Universidad de Chile. Doctor en Ciencias de la Computación, Universidad de Lugano, Suiza; Diplôme d'études approfondies, University of Caen, Francia; Maîtrise d'informatique, University of Caen, Francia; DEUG and Licence, informatique (Bachelor), University of Caen, Francia. Líneas de investigación: Lenguajes de Programación, Ingeniería de Software.

rrobbes@dcc.uchile.cl



JOCELYN SIMMONDS

Profesora Asistente Departamento de Ciencias de la Computación, Universidad de Chile. Doctora en Computer Science, University of Toronto, Canadá; Master of Science en Computer Science, Vrije Universiteit Brussel, Bélgica y École Des Mines de Nantes, Francia. Líneas de Investigación: Análisis y Diseño de Software, en especial aplicaciones Web y móviles; Validación y Verificación de Sistemas; Educación Apoyada con Tecnología, y cómo atraer mujeres a Ingeniería y Ciencias.
jsimmond@dcc.uchile.cl

MOTIVACIÓN

No hay un consenso general acerca de cómo clasificar a las empresas de software según su tamaño. En este artículo llamaremos microempresa a aquellas con hasta 10 empleados, pequeñas a las que tienen entre 11 y 50 empleados, medianas si tienen entre 51 y 100, y grandes a aquellas con más de 100 empleados. La principal fuerza laboral de la industria del software mundial está concentrada en las micro, pequeñas

medianas empresas de software, y particularmente en las dos primeras. Diversos reportes indican que en Chile más del 40% de las empresas de software tiene menos de 10 empleados (o sea, son microempresas) [Inn03], en Brasil el 70% corresponde a micro o pequeñas empresas [Wan06], en Canadá [Lap08] y en Estados Unidos [Fay00] las microempresas representan el 50% y el 78% de las organizaciones desarrolladoras

de software respectivamente. En el caso de Estados Unidos, el 95% de ellas tiene menos de 50 empleados [Ara10].

La realidad de las micro y pequeñas empresas de software es diferente a la de las medianas, y esta diferencia es aún mayor en relación con las grandes. Las primeras tienden a ser más vulnerables, pues usualmente tienen poco capital de riesgo y, por lo tanto, están obligadas a aceptar casi cualquier proyecto que se les presente, independientemente de su nivel de riesgo y del beneficio económico potencial que les representa.

Dado que la mayoría de las empresas nacen como micro o pequeñas organizaciones, cuando éstas llegan a ser medianas es porque ya están consolidadas. La mayoría de éstas hace software para nichos específicos y tienen procesos y personal estables. Esta realidad también está presente en las grandes empresas de software; sin embargo estas últimas son las únicas capaces de enfrentar el desarrollo de proyectos grandes, dada su capacidad financiera y de desarrollo de software. Esto también les permite escoger los proyectos que les son más atractivos, tanto desde el punto de vista tecnológico como económico.

En este sentido, vemos que la mayor parte de la industria de software chilena, o sea las micro y pequeñas empresas, es vulnerable. Por lo tanto, si se pretende que la industria del software tenga un impacto país, es en estas empresas donde se deben concentrar los mayores esfuerzos. Si consideramos que el desarrollo de software es una actividad transversal que permite que diversas áreas puedan brindar más y mejores servicios (por ejemplo, salud, sector productivo, seguridad, comercio y ocio), la mejora de la industria del software redundará directamente en la mejora de estos servicios y, por lo tanto, en el bienestar de nuestra sociedad.

En las siguientes secciones se analizan los diferentes aspectos vinculados al desarrollo de software, y los desafíos que enfrenta la industria en estos ámbitos. La información provista en este artículo fue obtenida principalmente a partir de



un estudio reciente, sobre las micro, pequeñas y medianas empresas (PyMEs) de software chilenas, el cual se desarrolló como parte del proyecto GEMS¹.

LOS PROYECTOS A REALIZAR Y LA CAPACIDAD DE ELEGIR

Un estudio realizado por Quispe y otros [Qui10] en micro y pequeñas empresas de software chilenas, muestra que más del 80% de los proyectos que ellas realizan parten con objetivos poco claros y, por lo tanto, con un riesgo muy alto. Esto se ve reflejado también en la baja efectividad que tienen las soluciones que éstas implementan (solamente el 25-30% terminan siendo efectivas).

La poca claridad en el alcance y objetivos de los proyectos se debe a que muchos de estos nacen a partir de un problema que se requiere solucionar con urgencia. Por lo tanto, el cliente no tuvo tiempo suficiente de pensarla en profundidad y detalle, y en consecuencia, la solución se va descubriendo durante la ejecución del proyecto. Por otra parte, es poco probable que las micro y pequeñas empresas de software conozcan bien el nicho de negocio en el que se enmarca el problema del cliente, por lo tanto generalmente es poco lo que éstas pueden hacer por ayudar a descubrir la solución al problema planteado. Dada la premura que usualmente hay por contar con la solución, los proyectos terminan desarrollándose contra reloj, con poco análisis y poca ingeniería. Lamentablemente, estas empresas de software tienen una casi nula capacidad de elegir sus proyectos, por lo que a pesar de las condiciones adversas, igual deben aceptar desarrollarlos. De otra forma ponen en riesgo su supervivencia. En un reciente estudio comprobamos que alrededor del 30% de las micro y pequeñas empresas tienen menos de cinco años de antigüedad, mientras que un poco más del 30% tiene entre cinco y diez años. Esto indica que no muchas pequeñas empresas subsisten fácilmente en el tiempo.

En el caso de las empresas medianas la situación es usualmente distinta, pues éstas trabajan en nichos o en productos específicos sobre los cuales tienen un conocimiento más profundo. Por lo tanto, sus proyectos tienden a ser menos riesgosos y además estas empresas tienen cierta capacidad de elegir, no sólo porque ya han construido una cartera de clientes, sino porque tienen respaldo financiero. Algo similar sucede con las empresas grandes, que claramente tienen una posición privilegiada en el mercado de software actual. Ellas son más apropiadas para abordar grandes proyectos, que les aseguran la sustentabilidad por varios años.

LOS PROCESOS DE DESARROLLO

Aproximadamente el 80% de las PyMEs de software declaran tener definido su proceso organizacional de desarrollo de software, aunque sólo el 50% dice aplicarlo. Las empresas de menos de cinco años desarrollan proyectos de tres meses en promedio, y las de más de cinco años trabajan en proyectos un poco más grandes (cinco meses en promedio). Estos desarrollos están enfocados esencialmente en la construcción del software en sí, reduciendo al máximo posible las actividades relativas a la gestión del proyecto. Esta realidad fue constatada durante el desarrollo del proyecto Tutelkán [Val10], el cual buscó ayudar a las PyMEs de software chilenas a definir sus procesos organizacionales.

LOS DESARROLLADORES Y LOS EQUIPOS DE TRABAJO

De acuerdo al estudio recientemente realizado, prácticamente el 50% de los desarrolladores tiene entre 25 y 30 años, y el 30% tiene entre 30 y 35 años. Por lo tanto son gente joven, con poca experiencia, pero con conocimiento sobre las nuevas tecnologías. Estas personas cuestan me-

nos, son relativamente fáciles de reclutar y tienen gran entusiasmo por surgir profesionalmente.

La cantidad de graduados de carreras de Computación e Informática en Chile parece estar por debajo de las necesidades del mercado. Este déficit de desarrolladores ha atraído a muchos profesionales de países vecinos.

Usualmente, cuánto más chica es la empresa, menos puede pagar por sus desarrolladores, y por lo tanto, la selección de personal que puede realizar es más acotada y menos exigente [Cole05, Mond05, Rich07]. En este sentido, las micro y pequeñas empresas son las más desfavorecidas, dada su limitada capacidad financiera.

En general estas empresas declaran que sus trabajadores participan en un solo proyecto a la vez, aunque aproximadamente el 25% indica que pueden llegar a trabajar hasta en cuatro proyectos en paralelo. Esto atenta contra la productividad de los trabajadores y también de la empresa.

LOS PRODUCTOS DE SOFTWARE

La calidad de los productos de software depende de muchos factores. Sin embargo, las empresas de nicho o bien aquellas que trabajan en productos puntuales tienen ventajas importantes respecto a las que desarrollan software para un ámbito más general, como es el caso de las micro y pequeñas empresas de software. Las empresas de nicho usualmente cuentan con el conocimiento tanto del propio del negocio, como de los aspectos técnicos requeridos para construir el producto. Esto les permite desarrollar software sólido desde el punto de vista técnico, y efectivo en la resolución de los problemas abordados.

Por su parte, las empresas que trabajan sobre productos específicos también cuentan con esta ventaja comparativa, pero además tienen la posibilidad de ir mejorando sus productos en forma evolutiva, lo cual hace que sus proyectos tengan baja incertidumbre y alto impacto.

¹ <http://www.dcc.uchile.cl/gems/>



La otra cara de la moneda la constituyen los productos desarrollados por las micro y pequeñas empresas que desarrollan software para cualquier nicho de mercado, dado que no tienen la capacidad de elegir sus proyectos. Estas empresas usualmente cuentan con conocimiento técnico, pero no del ámbito de negocio en el que se insertan sus proyectos, ya que estos ámbitos son muy diversos. Esto implica que aunque sus productos sean técnicamente buenos, es poco probable que impacten en forma significativa las organizaciones que reciben dichos productos. Si a esto le sumamos el escaso margen de tiempo y presupuesto con las que estas empresas operan, es muy difícil poder garantizar la calidad del producto final obtenido. Dado que este tipo de empresa son la mayor parte de la industria del software tanto en Chile como en el mundo, es explicable entonces por qué la mayoría de los desarrollos de software son tan poco efectivos.

LAS OPORTUNIDADES PARA GENERAR UN IMPACTO PAÍS

Hay diversas oportunidades de mejora de la fuerza laboral en el ámbito del desarrollo de software. Sin embargo, la falta de políticas gubernamentales en este sentido hace que cada empresa busque la forma de solucionar sus propios problemas, utilizando métodos *ad-hoc* que no aportan demasiado a la mejora de la industria de software local.

Una forma posible de aumentar el número de graduados en carreras de Computación e Informática, sin sacrificar la calidad de los mismos, radica en focalizar la formación de los mismos, reduciendo así la duración de los programas tal como se ha hecho en Europa. Otra posibilidad es fomentar la educación continua, facilitando la actualización periódica y focalizada de nuevos conocimientos, por parte de desarrolladores que podrían estar comenzando a salir del mercado por la obsolescencia de su conocimiento. Dada

la velocidad a la que avanza la tecnología, los períodos de caducidad del conocimiento se vuelven cada vez más cortos.

En este sentido existe una oportunidad real de robustecer la industria local de software y generar un impacto país. Sin embargo, esto requiere asumir compromisos por parte de las empresas, los profesionales involucrados, el gobierno y las instituciones que forman a estos profesionales. Un ejemplo a seguir es la iniciativa llevada a cabo por Irlanda, la cual generó en menos de quince años una industria de software capaz de compe-

tir a nivel global [Ryan15], pero en nichos especializados. En ese sentido, enfocarse en el desarrollo de software para nichos específicos podría ayudar a la industria chilena a mejorar su competitividad, tanto en la región como a nivel global.

Enfocarse en la implantación y mejora de procesos, pero de cara a las PyMEs, podría traer grandes beneficios a la industria local. Muchas empresas buscan alcanzar certificaciones internacionales que están pensadas para com-

pañías que compiten a nivel global, y que no les ayudan a las PyMEs a ser más competitivas. Por lo tanto, cualquier esfuerzo invertido en ese sentido tendrá poco o ningún impacto en la supervivencia de la mayor parte de la industria chilena del software. Es tiempo de comenzar a mirar hacia adentro, y también a nuestros vecinos, buscando identificar recetas para el éxito que puedan ser replicadas en Chile. En este sentido, la academia, la industria y el gobierno tienen a la mano la oportunidad de trabajar juntas para generar un impacto país. ■

REFERENCIAS

- [Ara10] J. Aranda (2010) Playing to the strengths of small organizations. 1st Workshop on Requirements Engineering in Small Companies (RESC2010), at the 16th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'10), Essen, Germany, 2010.
- [Ben08] J.M. Benavente, C. Kulzer. Creación y destrucción de empresas en Chile. Estudios de Economía [online]. 2008, vol.35, n.2, pp. 215-239.
- [Cole205] K. Coleman, P. Larsen, M. Shaw, M.V. Zelkowitz. Software Process Improvement in Small Organizations: A Case Study. IEEE Software, Vol. 22, No. 6, pp. 68-75, Nov./Dec. 2005.
- [Gech08] GECHS AG y Escuela de Ingeniería de la Universidad de Viña del Mar. Sexta versión del Estudio Diagnóstico de la Industria Nacional de Software y Servicios, 2008.
- [Inn03] Programa CHILE INNOVA. Diagnóstico de la industria de tecnologías de la información en Chile 2003. White Paper, 2003, URL: <http://www.estrategiadigital.gob.cl/files/DIAGNOSTICO%20INDUSTRIA%20TIC%202003.pdf>
- [Lapo08] C.Y. Laporte, S. Alexandre, A. Renault. Developing International Standards for Very Small Enterprises. IEEE Computer, Vol. 41, No. 3, pp. 98-101, Mar. 2008.
- [Mond05] O. Mondragon. Addressing infrastructure issues in very small settings. First International Research Workshop for Process Improvement in Small Settings, Pittsburgh, PA, 2005.
- [Quis01] A. Quispe, M. Marques, L. Silvestre, S.F. Ochoa, R. Robbes. Requirements Engineering Practices in Very Small Software Enterprises: A Diagnostic Study. Proceedings of the XXIX International Conference of the Chilean Computer Science Society, pp. 81-87. IEEE Press. Antofagasta, Chile, Nov. 15-19, 2010.
- [Rich07] I. Richardson, C. G. von Wangenheim. Why Are Small Software Organizations Different? IEEE Software, Vol. 24, No 1, pp. 18–22, Jan./Feb. 2007.
- [Ryan15] K. Ryan. The experience of the Irish Software Sector – Successes, Failures and Lessons Learned. Keynote Speech at the Computer Science Department of the University of Chile. May 29th, 2015.
- [Vald10] G. Valdés, H. Astudillo, M. Visconti, C. López. The Tutelkan SPI Framework for Small Settings: A Methodology Transfer Vehicle. Proceedings of EuroSPI'2010: Communications in Computer and Information Science, Vol. 99, pp. 142-152, 2010.
- [Wang06] C. G. von Wangenheim, A. Anacleto, C. F. Salviano. Helping Small Companies Assess Software Processes. IEEE Software, Vol. 23, No 1, pp. 91 – 98, Jan./Feb. 2006

DATA SCIENCE APLICADA AL DESARROLLO DE PRODUCTOS

Quizás algunas de las referencias que se mencionan en este artículo sólo sean memorables para los que nos peinamos con canas blancas. Desde ya le pido disculpas a las jóvenes mentes que lean este texto. Aun así deseo aportar a la conversación sobre esta interesante historia que incluye integración de los *data scientists*, utilización de datos abiertos y la evolución de plataformas tecnológicas, entre otros entremeses.





PABLO GARCÍA

Gerente de Investigación y Arquitecto Jefe de Telefónica Investigación y Desarrollo de Chile. Anteriormente se desempeñó como arquitecto principal en Microsoft durante más de una década y anteriormente como líder técnico y director de la empresa Urudata. Es profesor de Programación, Estructuras de Datos y Algoritmos para el Programa de Desarrollo de las Ciencias Básicas de Uruguay. Es Máster en Computación y sus áreas de interés principales son la automatización de procesos y el procesamiento de secuencias planas de ADN.

pablo.garcia@telefonica.com / [@pc_garcia](https://twitter.com/pc_garcia)

Hace algunos años vi una presentación de Jim Gray –un científico que la mayoría de la gente recuerda por sus trabajos en transacciones distribuidas o por el ACID test–, donde trataba de difundir un concepto que había expuesto en la Academia de Ciencias Americana, en un discurso ante el Congreso. En esa charla, él hablaba que la ciencia había pasado por varios paradigmas.

El primero, que supuso un nuevo mecanismo de avance sustancial para la ciencia, lo estableció Galileo con el establecimiento de la ciencia experimental. El segundo es la de la ciencia teórica. Ésta aparece con el trabajo de Newton sobre los planetas, el que establece que en su movimiento de traslación estos recorrian áreas iguales en tiempos iguales.

Luego, la teoría atómica nos ayudó a establecer el paradigma de las simulaciones. En ello primero se teorizaba, luego se construía una simulación y posteriormente se la contrastaba con el experimento. Finalmente, ya en nuestros días, llegamos a un nuevo paradigma, caracterizado por el volumen enorme de datos. Fuentes como el secuenciamiento de ADN, la ciencia del clima y la telemetría de muchísimos dispositivos conectados, hace que la aplicación de técnicas computacionales de descubrimiento de patrones, establezca un nuevo paradigma para el avance de la ciencia. A este nuevo paradigma podemos llamarle la Ciencia de los Datos o en su acepción anglosajona: *Data Science*.

Ese nuevo paradigma ya traspasó las fronteras de la ciencia y hoy vemos una enorme curiosidad dentro de las empresas por mutar su estrategia de *datawarehousing* a un modelo que los aleje de los costosísimos procesos de Extracción, Trans-

formación y Carga, eliminando la Transformación y cargando los datos en bruto, aplicando luego fuerza de procesamiento para transformarlos al momento de analizarlos.

Como todo en la vida, este nuevo paradigma tiene ventajas y desventajas. La principal utilidad es que la tecnología que se usa para estos procesos, en el día de hoy, es un *commodity*. También se ha convertido en un *commodity* el tener enormes capacidades de almacenamiento y de procesamiento *on-demand*, a precios ridículamente bajos, aunque esto es la parte menos importante de la ecuación.

Lo que hoy complica es que el equipo de *Business Intelligence* (BI) no puede hacer la transición de ordenador de la información a *data scientist*, siendo éste el dilema en que se encuentran hoy las organizaciones: todos miran los reportes de las consultoras como Gartner, y muchos desean abordar este tema dentro de su organización, pero se encuentran con que los *data scientists* son pocos y a la mayoría no les gusta hacer reportes sofisticados de marketing.

DATA SCIENTISTS INTEGRADOS AL NEGOCIO

Para aplicar *Data Science* hay que dejar claro que esto implica muchos requerimientos, pero podemos establecer algunas bases mínimas, tales como: saber procesar grandes volúmenes de datos,

saber analizarlos en forma rigurosa, comprender la relevancia estadística de los resultados obtenidos y saber del negocio para poder interpretar los resultados.

Hace unos años atrás hice un Magíster en Bioinformática, y cuando tomé las materias electivas del área biológica no me pareció relevante tomar Evolución. Sin embargo, al avanzar en la maestría, me pasó que los biólogos descartaban rápidamente resultados aplicando la teoría de la evolución con un razonamiento que resultaba muy difícil de formalizar. Era su conocimiento de cómo cambian organismos y poblaciones, influenciados por la evolución, lo que les permitía rápidamente descartar un grupo de resultados que parecían matemáticamente relevantes, y concentrarse en los que tenían sentido biológico.

Me refiero con esto a que, para poder aplicar técnicas y conocimiento de estadísticas y *machine learning* en el procesamiento de bases de datos de un negocio y entregar algo útil como salida, se necesita conocer del negocio en particular. No basta simplemente con traer un experto y que él realice el reporte, es necesario integrarlo a la organización y empaparlo en el negocio para que aparezcan los mejores resultados.

Podemos tentarnos a armar un equipo multidisciplinario, pero con esto no alcanza. Por ello, de las cosas que estamos realizando en el Centro de Excelencia Internacional de Investigación y Desarrollo de Telefónica (en alianza con Corfo y la Universidad del Desarrollo) además de integrar equipos de *data scientists* con especialistas en minería, agricultura de precisión y ciudades inteligentes, es a no escatimar recursos para que nuestros *data scientists* aprendan en profundidad el contexto en el que están desarrollando su trabajo.

Existen enormes oportunidades de hacer minería de procesos y aplicar diversas técnicas de minería de datos, pero para hacer la diferencia profunda dentro de la organización se necesitan recursos humanos, tiempo para dominar las técnicas y madurar el conocimiento del negocio. Todavía estamos en etapas muy iniciales, lejos de recoger los enormes beneficios que promete el estado del arte de la tecnología.

LOS DESAFÍOS EN DATA SCIENCES

Hay muchísimo por hacer en la tarea de cimentar la importancia de la aplicación de data science en el desarrollo tecnológico. En Telefónica I+D Chile encontramos a diario lindos problemas para resolver: desde la falta de arquitecturas que soporten implementar soluciones confiables y escalables, pasando por la configuración de redes de sensores, hasta la aplicación masiva de técnicas de machine learning que hace muchos años están consolidadas, pero que todavía no están adecuadamente implementadas en las plataformas de código abierto que usamos en nuestros proyectos.

Hoy en día, los grandes proveedores de cloud computing están implementando soluciones interesantes para integrarlas al gran volumen de datos que están acumulando. Por ejemplo, el trabajo que hace Microsoft con Revolution Analytics y Azure Machine Learning, son extremadamente prometedores, así como las soluciones de Google y Amazon. Sin embargo, todas ellas, a pesar de lo interesante y prometedoras, todavía son plataformas que carecen de modelos que se adapten automáticamente a la casuística de una organización particular y que sean efectivas para la misma.

En la actualidad se requiere de alguien altamente especializado para que genere un modelo predictivo eficiente, que haga la diferencia para el negocio de la organización. En este sentido, aparecen startups prometedoras que venden servicios para determinadas verticales y protegen sus modelos detrás de la venta del servicio, llevando los datos del cliente a su entorno de procesamiento. No obstante, deberíamos ver una evolución en el sentido inverso, donde se puedan portar y adaptar los modelos en el entorno de ejecución del cliente, llevando el procesamiento a los datos y sirviéndose de la nube como una fuente más de información, donde las empresas del mismo rubro aportan anónimamente sus datos para evolucionar los modelos predictivos y donde se integran fuentes de datos que tiene sentido procesarlas en la nube en su propia fuente, tales como dato de redes sociales, por ejemplo.

EVOLUCIÓN DE PLATAFORMAS TECNOLÓGICAS

Estamos viendo madurar y evolucionar diversas plataformas tecnológicas. En particular, hace algunos años, tecnologías como Hadoop competían con otras plataformas, mientras los grandes de la industria desarrollaban sus propios modelos de computación distribuida, y hoy en día esa discusión ya es cosa del pasado. Así como también lo fue en un momento la existencia de diversos protocolos de redes LAN, hasta que se impuso el TCP/IP, hoy pasa algo similar con Hadoop y todo el ecosistema que corre sobre él: Storm, Spark, cascading, Scala, Hive, Hbase, etc. Vemos varias distribuciones consolidadas, la separación del almacenamiento de los data nodes, la evolución basada en yarn, todas las cuales permiten crear soluciones de órdenes de magnitud más eficientes y pensar en soluciones que operen en tiempo real.

Estas distribuciones permiten a personas con poco entrenamiento aprender la parte de manipulación de datos de Hadoop con sólo bajar una imagen de máquina virtual, trabajar con una solución que se despliega muy rápido y que permite, en las versiones de nube, almacenar los datos con el cluster apagado, y prenderlo para procesar los datos sólo cuando uno lo necesita (en algunos casos, pagando por minuto procesado).

Esta tecnología se ha convertido en algo barato y abundante, además de tener un ritmo de evolución vertiginoso. La plataforma va a seguir evolucionando más y más rápido, incluso de forma más apresurada de lo que las organizaciones pueden digerir. Tenía un profesor en la universidad que solía decir “la historia acelera”. Creo que es muy cierto.

La abundancia de capacidad de cómputo y de almacenamiento a costos muy bajos en las soluciones cloud, más el alineamiento de la comunidad académica y la industria a nivel mundial, para evolucionar sobre plataformas como Hadoop,

genera, sin dudas, espacios para que aparezcan las soluciones que logren explotar esos datos, aportando con una diferencia significativa al negocio.

Sin embargo, si vuelvo a mis tiempos de arquitecto de una organización, hay dos decisiones de arquitectura, tres fundaciones sobre las cuales construiría la estrategia de datos de la organización y que me parece son elementos previos de los cuales carecen la mayoría de las organizaciones hoy.

Primero, almacenaría todos los datos que generan los sistemas de la organización en su forma original, sin ningún tipo de transformación y en la forma más sencilla posible. Todo, no se descarta ni resume, nada.

Segundo, no permitiría que se comprara ningún sistema que no entregue, en forma de datos abiertos (datos de los cuales se puede interpretar su significado en forma integral), todos y cada uno de los datos que capture de mi organización. Y éste, aunque parezca menor, es un punto crítico hoy en Chile. Al realizar un proyecto a una minera importante y cuando pide la telemetría de algo tan simple como una pala mecánica y dice que no, sólo el proveedor puede leer los datos de la máquina, si se conecta algo a la interfaz de la computadora de la máquina se pierde la garantía. Y en último punto, no sólo se requiere de hardware, sino también es indispensable que todo esté en la nube.

En este tiempo donde el más desinformado de los gerentes entiende que hay valor en los datos, en el concepto de Big Data, la situación de datos abiertos es inadmisible y es una decisión innegociable que tiene que imponer la arquitectura de la información; no aceptar islas de datos dentro de la organización, no aceptar proveedores que se quedan con nuestros propios datos y no nos los entreguen.

Antes hablaba de lo que se necesita para hacer data science. En un nivel más básico, para hacer machine learning se necesitan datos, patrones existentes en los datos y que estos no sean formulables en forma matemática. Bueno, estas son las tres bases para hacer machine learning, pero la más básica de todas es... ¡tener datos!

LA EXPERIENCIA EN TELEFÓNICA I+D

Nosotros trabajamos con una visión muy pragmática y partimos de problemas reales, enfocados principalmente en los desafíos referidos de minería, agricultura y urbanismo que enfrentan nuestros clientes (Market Pull). Sobre esos inconvenientes, nos preocupamos de capturar toda la información existente y de integrar sensores para capturar y poner toda la información en una plataforma de Internet of Things (IoT).

Esto implica que cada objeto instrumentado tendrá una existencia física en el mundo real y otra virtual, contextualizada y conectada con la instancia física, existiendo dentro de nuestra plataforma. Adicionalmente, capturamos toda la historia de cada elemento y agregamos la información de sistemas existentes y fuentes externas que sean relevantes para darle contexto al elemento.

Podemos tener la telemetría de un camión, pero es relevante para el contexto tener, por ejemplo, la condición climatológica en el lugar por donde circula el camión. Los equipos de desarrollo trabajan en los problemas de integración, visualización de los datos de los sensores, integraciones diversas y hasta el procesamiento de eventos complejos de primer nivel.

Los datos nos hablan. Por ejemplo, si aumentan las señales alternadas en un período de tiempo es indicación de que el sensor está haciendo mal contacto; un aumento repentino de consumo de energía implica una manguera de presión con pérdidas; un aumento sostenido del consumo de energía, en general indica falta de lubricación, etc.

LA APERTURA DE LOS DATOS

Otro tema muy importante que mencioné al comienzo es la disponibilidad de los datos. Respetando en primera instancia los elementos de pri-

vacidad de datos, siempre hay espacios para compartir información.

Esta arista sobre compartir los datos es muy clara y común en el caso de las ciudades: hay múltiples iniciativas de ciudades digitales y todas tienen un componente de datos abiertos. Sin embargo, a nivel empresarial no existe casi apertura y si bien esto obedece en general a políticas internas de seguridad de la información, hay datos que son valiosos por su transversalidad y porque al integrarlos, esto genera redes de información que aportan más valor que las islas (nodos) que la componen. Por ejemplo, si tengo una estación agroclimatológica, debería ser fácil para mí compartir los datos de la misma y para otros, encontrar esos datos y consumirlos, porque el poder de ajuste de predicciones que da el tener las estaciones de mis vecinos y de los vecinos de mis vecinos, es enorme.

Si somos capaces de ver el valor de la información agregada y compartida, aparecerá la necesidad de tener una plataforma transversal, e idealmente abierta, en donde agregarla y explotarla. Por ello, en Telefónica I+D montamos todas las soluciones sobre la plataforma FIWARE. Esta plataforma es creada como proyecto abierto por la comunidad económica europea, y si bien hoy la discusión de software comercial o abierto es un tema casi del pasado, si es relevante el tema de tener la facilidad de manejar datos abiertos, es decir, que mis datos estén en un formato que los pueda consultar e interpretar tanto un humano, como una máquina, con acceso simple y basado en tecnologías estandar (acceso REST, transmisión sobre http, encapsulación en json, etc).

Para nosotros resulta esencial el poder compartir los datos abiertos que dispongamos. Si queremos tener un ecosistema de IoT activo y dinámico, es necesario que podamos darle de forma sencilla acceso a los datos a los emprendedores. El poder de entregar permiso de lectura al usuario "Juan" sobre todos los sensores del tipo "estación de clima" del grupo "Región de Coquimbo", y que con esta simple acción le demos todas las facilidades de consulta y procesamiento, es una de las capacidades "out of the box" que queremos tener en corto plazo con esta plataforma. ■

RED CHILENA DEL MEJORAMIENTO DEL PROCESO DE SOFTWARE, SPIN- CHILE A.G.

Entre 1995 y 1997 la Corporación de Investigación Tecnológica (INTEC), en el contexto del proyecto "Herramientas de productividad para la industria de software" realizó las primeras acciones nacionales conducentes a mejorar la calidad de desarrollo de software nacional. Una de sus primeras actividades fue realizar una gira para conocer el estado del arte internacional, momento en el cual se conoció de la existencia de SPIN (Software Process Improvement Network) creándose en el año 1995 el organismo nacional, denominado "Club CMM".





ERIKA MADARIAGA

Ingeniero Civil en Informática, Universidad de Santiago de Chile ; Magíster en Educación Basada en Competencias, Universidad de Talca; Diplomado en Gestión de Negocios, Pontificia Universidad Católica de Chile. Actualmente Directora de la Escuela de Computación e Informática de la Universidad Bernardo O'Higgins, y Directora Secretaria de SPIN-Chile A.G. Con más de 15 años de experiencia docente y directiva en diversas instituciones de educación superior. Especialista en formación y gestión por competencias en el área TI.
erika.madariaga@ubo.cl

Dentro de las actividades de ese proyecto estuvo el acuerdo con el CRIQ de Canadá para usar una herramienta de evaluación de riesgos de desarrollo S:Prime, evaluaciones piloto y capacitación de consultores en la herramienta. Este proyecto finalizó en noviembre de 1997 con la realización del "Primer Simposio Latinoamericano de Calidad y Productividad en Desarrollo de Software".

En sus inicios esta agrupación de profesionales, Club CMM, comenzó a funcionar bajo el alero de INTEC-Chile, y su principal objetivo fue introducir técnicas y métodos útiles a los desarrolladores de software, apuntando a la mejora de sus competencias. En 1996 esta agrupación se integró formalmente a la red mundial de SPIN liderado por el SEI (Software Engineering Institute) de la Universidad de Carnegie Mellon, y pasó a llamarse SPIN-Chile, cuyo objetivo fue compartir conocimientos y experiencias, en el ámbito chileno, y relacionarse con quienes trabajan por similares intereses en el mundo.

Al término del proyecto en 1997, SPIN-Chile se separó de INTEC y comenzó a funcionar bajo el alero de la Universidad de Chile, dado que no tenía personalidad jurídica. En enero de 2003, luego de cumplir con los procesos legales vigentes de la época, el grupo se constituye y consolida como una asociación gremial, pasando a ser denominado SPIN-Chile A.G.

Grupos SPIN existen en todo el mundo, bajo un modelo propuesto y respaldado por el SEI. Cada SPIN tiene una misión y visión diferentes, basadas en las necesidades de la comunidad donde se desarrollan. La característica común a todos ellos es el aporte de tiempo y recursos de sus miembros para el desarrollo de las actividades y el logro de sus objetivos¹.

SPIN-Chile A.G.² actualmente es una agrupación de empresas, universidades y profesionales en torno

al mejoramiento de los procesos en las prácticas de Ingeniería de Software en el mercado TIC chileno, que conforman el capítulo local de SPIN, realizando sus actividades regulares en Santiago de Chile.

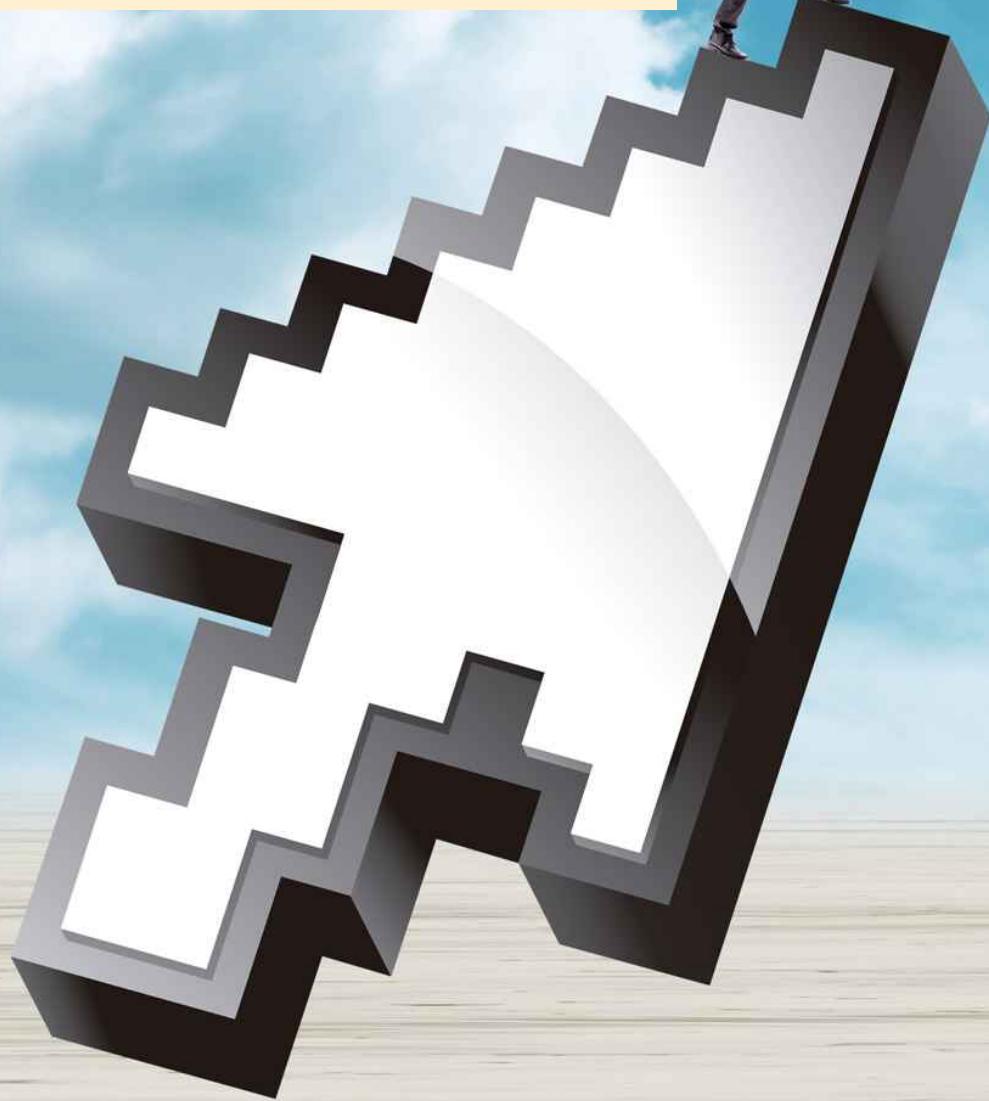
SPIN-Chile A.G. organiza anualmente un conjunto de actividades tendientes a promover la interacción entre sus miembros, y la orientación está relacionada con las características, capacidades y requerimientos de cada uno de los tipos de socios que congrega. Las principales actividades son: tutoriales, reuniones presenciales, mesas redondas, seminarios, cursos, talleres, workshop, entre otros. Todas estas actividades son organizadas con la colaboración de los distintos miembros que conforman la red, y el objetivo principal es mostrar el estado del arte, buenas prácticas, experiencias de empresas en el proceso de mejoramiento de la calidad, éxitos y fracasos. Destaca en 2007, la realización del evento internacional realizado en Santiago de Chile, Software Engineering Process Group Latin America (SEPLGA 2007) organizado por SPIN-Chile A.G., European Software Institute (ESI), y patrocinado por Software Engineering Institute (SEI), el cual contó con la participación de profesionales de distintos países de Latinoamérica y conferencistas de reconocimiento mundial.

Actualmente, SPIN-Chile A.G. está potenciando sus alianzas con universidades e instituciones afines, y desarrollando nuevos proyectos, entre los cuales destaca el realizar estudios que permitan entregar a la comunidad chilena antecedentes locales sobre la calidad en el desarrollo de las soluciones, sistemas y servicios de software. Lo importante es que estos estudios estén debidamente respaldados y que permitan sustentar con información actualizada decisiones de mejoras en las prácticas de ingeniería de software y en la enseñanza impartida a sus profesionales en formación, en pos de la excelencia de la industria TIC en Chile. ■

¹ <http://www.sei.cmu.edu/spin/>

² <http://www.spin-chile.cl/>

POSTÍTULOS EN TECNOLOGÍAS DE LA INFORMACIÓN





MAGÍSTER EN TECNOLOGÍAS DE LA INFORMACIÓN UNIVERSIDAD DE CHILE: POSTGRADO PARA PROFESIONALES



JOSÉPINO

Profesor Titular del Departamento de Ciencias de la Computación (DCC), Universidad de Chile; coordinador académico del Magíster en Tecnologías de la Información.

jpino@dcc.uchile.cl

El Departamento de Ciencias de la Computación de la Universidad de Chile ofrece desde 2004 el Programa de Magíster en Tecnologías de la Información (MTI). Este Programa está orientado a profesionales del área que estén trabajando en empresas e instituciones. Se ofrece en un formato vespertino y los cursos tienen un marcado énfasis en temas prácticos, tecnológicos y de educación continua.

La mayor parte de los cursos del MTI son electivos, clasificados en cuatro áreas: Gestión Informática, Ingeniería y Calidad de Software, Ingeniería de Datos, y Seguridad Computacional. El resto de los cursos (obligatorios) se refieren al desarrollo del Proyecto de Grado final. La duración estimada de los estudios es de dos años.

El Programa tiene otra característica interesante: al cursar un cierto número de cursos de una cierta área se obtiene un Diploma de Postítulo. La mayoría de los alumnos del Programa obtiene dos Postítulos de I Departamento como beneficio adicional de seguir el MTI. Los profesores del Programa son académicos de jornada completa del Departamento y profesores de jornada parcial que laboran en empresas o instituciones.

Los requisitos de postulación son: estar en posesión de una Licenciatura en Computación o equivalente, tener buenos antecedentes académicos, tener alguna experiencia laboral.

Mayores antecedentes pueden encontrarse en:

http://www.dcc.uchile.cl/magister_ti y las consultas pueden dirigirse a: magisterTI@dcc.uchile.cl

POSTÍTULO EN INGENIERÍA Y CALIDAD DE SOFTWARE, UNIVERSIDAD DE CHILE



MARÍA CECILIA BASTARRICA

Profesora Asociada del Departamento de Ciencias de la Computación (DCC), Universidad de Chile; coordinadora académica Diplomados en Ingeniería de Software.

cecilia@dcc.uchile.cl

El Postítulo en Ingeniería y Calidad de Software se dicta en el Departamento de Ciencias de la Computación (DCC) de la Universidad de Chile desde 2004, y cuenta ya con más de 120 titulados. Solamente se admite para este Programa a personas que tengan un título de Ingeniero Civil en Computación, o bien algún título afín y varios años de experiencia comprobable en el desarrollo de software.

Este Postítulo se divide en dos módulos: Ingeniería de Software y Calidad de Software, cada uno de los cuales constituye un diploma por sí mismo. Los cursos de este programa son todos válidos como parte del Magíster en Tecnologías de la Información que imparte el DCC.

Gran parte de los alumnos son ingenieros que obtuvieron su título hace más de diez años y muchos de ellos hace ya años que no programan ni están familiarizados con nuevas tecnologías y metodologías de desarrollo de software. En la primera parte del Postítulo de Ingeniería y Calidad de Software –Ingeniería de Software- se actualizan los conocimientos de los alumnos en temas tales como análisis, diseño y programación orientada a objetos, arquitecturas de software, desarrollo de software basado en modelos o en aspectos, o métodos ágiles de desarrollo, entre otros. En los últimos años se ha agregado también un curso de programación Python que ha sido altamente valorado por los alumnos.

El módulo de Calidad de Software se centra tanto en la productividad de los procesos de desarrollo de software como en la calidad de los productos de software en sí. En cuanto a la definición de procesos de desarrollo de software, se introducen conceptos básicos, técnicas s de

definición y estándares internacionales propuestos en relación a este tema. En cuanto a la calidad del software, se centra en formas de realizar pruebas de la calidad del software, así como un taller de aplicación de todo el contenido impartido.

En términos generales, los alumnos que se han titulado de este Programa lo han encontrado muy valioso y útil para su vida profesional en cuanto a los contenidos impartidos, pero también reconocen que requiere gran dedicación y estudio, ya que es muy exigente. ■



MAGÍSTER EN TECNOLOGÍAS DE INFORMACIÓN Y GESTIÓN (MTIG), PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE

El Programa de Magíster en Tecnologías de Información y Gestión (MTIG) nació en 2005 en el Departamento de Ciencia de la Computación (DCC) de la Escuela de Ingeniería de la Universidad Católica. La motivación fue poder transferir rápidamente habilidades técnicas y de gestión a ingenieros que trabajan directamente con tecnologías de información (TI), o ingenieros con responsabilidades de gestión en empresas donde esas tecnologías juegan un rol importante.



YADRÁN ETEROVIC

Profesor Asociado Departamento de Ciencia de la Computación (DCC), Pontificia Universidad Católica de Chile. Director del Postítulo en Gestión Informática (INGES) y coordinador docente del DCC UC.

yadran@ing.puc.cl



JAIME NAVÓN

Profesor Asociado Departamento de Ciencia de la Computación (DCC), Pontificia Universidad Católica de Chile. Director del DCC UC y del Magíster en Tecnologías de Información y Gestión (MTIG).

jnavon@ing.puc.cl

El sello del Programa es la dualidad tecnología-gestión. En un mismo curso suelen encontrarse ingenieros comerciales que saben poco de TI, ingenieros de software que saben poco de negocios e ingenieros eléctricos que saben poco de ambas cosas. El currículum es, por lo tanto, balanceado e incluye cursos técnicos sobre ingeniería de software o tecnologías de la Web, cursos de gestión estratégica o de personas, y cursos que unen ambas cosas, tales como Inteligencia de Negocios y Minería de Datos.



Los profesores del MTIG son en su mayoría profesores del DCC, pero también se cuenta con profesores part-time con amplia experiencia en la industria que aportan la otra componente.

El Programa incluye 24 cursos más una actividad de graduación en que los participantes tienen la oportunidad de aplicar sus nuevos conocimientos en un caso concreto. La mayoría de las veces, el tema para esta actividad surge de la empresa donde el alumno se desempeña profesionalmente y corresponde a un problema real.

En los diez años de vida del MTIG hemos graduado a cientos de profesionales que vuelven a sus respectivas empresas habiendo adquirido nuevas competencias y herramientas. Más relevante aún, vuelven a sus empresas con una nueva visión del rol crucial de las TI como habilitador de nuevos negocios. Gerentes de negocios aprenden, por un lado, sobre el tremendo poder de estas tecnologías, y, por otro, que se requiere conocimientos

técnicos, esfuerzo y tiempo para construir algo que finalmente parece relativamente simple desde la perspectiva del usuario.

Si bien el MTIG no ofrece especializaciones, en la práctica los estudiantes pueden tomar cuatro cursos del área de Ingeniería de Software (y hacer así una "miniespecialización"): Procesos de Desarrollo de Software, Taller de Ingeniería de Software I, Taller de Ingeniería de Software II, y Arquitectura de Sistemas. Estos cursos tienen una clara orientación al desarrollo de competencias por la vía del trabajo práctico (incluso, en clases) en grupos y en contextos de problemas, proyectos y aplicaciones reales, algo que los estudiantes valoran mucho: en los últimos tres años, estos cursos han tenido una inscripción promedio de casi veinte estudiantes cada uno. Por supuesto, los estudiantes complementan estos cursos con otros, en áreas tales como Bases de Datos, Tecnologías de la WWW o Gestión de Proyectos. ■

PROGRAMAS DE POSTÍTULO DEL DEPARTAMENTO DE INFORMÁTICA DE LA UTFSM



RAÚL MONGE

Director Departamento de Informática Universidad Técnica Federico Santa María (UTFSM), Valparaíso.
rmonge@inf.utfsm.cl

El Departamento de Informática creó en 1994 su primer Programa de Postítulo, un Diplomado en Ingeniería de Software. Actualmente, existen cuatro programas vigentes, a saber: en Gestión de Procesos de Negocios, Gestión de Proyectos, Base de Datos e Inteligencia de Negocios. Estos cuatro programas se describen a continuación.

a) Diploma en Business Process Management (DBPM)

Este Programa está orientado a profesionales en el área de Negocios y Tecnologías de la Información, con el propósito de formar especialistas en Business Process Management y en las TI que los soportan. El Diploma abarca el ciclo completo de la gestión de procesos; desde el alineamiento

estratégico, modelos de negocio, arquitectura de procesos, diseño de procesos de negocio, optimización de procesos, la implementación en entornos modernos de workflow, hasta la formulación de indicadores y el monitoreo en línea de la ejecución de los procesos integrados. Este Programa se inició en 2009, y ha logrado titular a 88 profesionales.

b) Diploma en Gestión de Proyectos en Tecnologías de la Información
Este Programa está orientado a profesionales que se desempeñan en distintos ámbitos de las TI, para formarlos en las mejores prácticas de dirección de proyectos y, al mismo tiempo, los enfrenta a casos donde estos conceptos deben ser aplicados para lograr un desarrollo exitoso. El Diploma se imparte desde 2007 y siempre ha tenido una muy alta demanda.

c) Diploma en Diseño y Administración de Bases de Datos

Este Programa se orienta a profesionales que se desempeñan en el campo de las tecnologías de la información y disciplinas afines, y que requieren especializarse en el área de Bases de Datos. El Programa los prepara mejor para enfrentar problemáticas propias de las organizaciones modernas, como el modelado de datos a partir de una especificación de requisitos, diseño lógico y físico de datos, administración de las bases de datos, integración entre sistemas operacionales y los sistemas de apoyo a la toma de decisiones, entre otros. Este Programa se inició en 2008 con el nombre de Diploma en Gestión de Datos, siendo actualizado en 2013 y recibiendo entonces el nombre actual.

d) Diploma en Inteligencia de Negocios

A partir de 2015, derivado de la actualización del Programa anterior en 2013, el Departamento de Informática ofrece este Programa, que se orienta a profesionales de las tecnologías de la información y disciplinas afines con el propósito de especializarlos en el área de la Inteligencia de Negocios. ■



ENTREVISTA A: ROBERTO OPAZO

Por Alejandro Hevia



Emprendimiento

DCC

Innovación



DCC



ROBERTO OPAZO

Ingeniero Civil en Computación y Magíster en Ciencias de la Ingeniería Industrial de la Universidad de Chile. Director ejecutivo y socio fundador de Khipu, una empresa que nació en Chile para competir en la industria de los medios de pago y que ya vendió su primera franquicia a Bolivia. Antes fue socio fundador y gerente general de Acepta.com, la empresa que partió en Chile con la firma y la factura electrónica. Durante su vida como emprendedor, también ha destinado tiempo a dictar las cátedras de "Seguridad de datos y firmas digitales" y "Emprendimiento en negocios digitales" en la Universidad de Chile. Inició su trayectoria como emprendedor después de haber sido empleado diez años en consultoras especializadas en la banca, el desarrollo de sistemas y la industria forestal. De estudiante cultivó el desarrollo de habilidades blandas en el teatro, como dirigente estudiantil, profesor auxiliar y consultor.

¿Seguiste la especialidad de Ciencias de la Computación y luego hiciste el Magíster en Ingeniería Industrial. Puedes resumir tu vida luego de la Escuela, contarnos un poco cómo llegaste a emprender?

Te respondo desde un poco antes. El DCC me cambió la vida en el primer semestre. Tuve clases con Juan Álvarez, en Introducción a la Computación, y ese fue el momento en que descubrí mi vocación por la Computación. Entré pensando en ser Ingeniero Civil Industrial y en ser político. Pero en ese semestre decidí ser "computín". Después me siguió gustando, si bien entremedio mantuve una "pata" en el Departamento de Ingeniería Industrial y hoy pertenezco a los exalumnos de ambos departamentos.

La verdad es que empecé a trabajar en el segundo año de Ingeniería. Trabajé tres años en el grupo de consultoría del Centro de Computación, luego tres años en investigación operativa en el grupo de Andrés Weintraub en el Departamento de Ingeniería Industrial de la Universidad de Chile. En el proceso, egresé y trabajé varios años. Y allí recién emprendí. Aunque no tenía tantos años de trabajo, ya tenía mucha experiencia. Mi primer emprendimiento estuvo asociado a mi tema de tesis, que era una solución al problema de asignación de horarios y fracasó por completo! Le mandé cartas a 400 colegios, muchos respondie-

ron pero quienes querían la solución no podían pagar por ella. Al final, aunque gasté plata y tiempo en ello, fracasó. Luego mi siguiente emprendimiento, por allá por 1998, fue algo que partimos en conjunto con Jorge Claro: un mall virtual. La idea era hacer un mall virtual grandote, pero eso también fracaso. Y en ese contexto fue cuando partimos Acepta.com, proyecto que era un plan de contingencia, que terminó siendo una historia de doce años para mí como emprendedor, donde transformamos a Chile en el primer país del mundo en emitir una factura electrónica en que la seguridad estaba dada por una firma electrónica. Después de Acepta.com partí Khipu.

Ya con tu experiencia, ¿qué conclusiones sacas respecto a qué te ayudó y que no te ayudó de la formación del Departamento de Ciencias de la Computación (DCC) y la Escuela de Ingeniería y Ciencias de la Universidad de Chile, en el proceso de emprender?

Mi respuesta es respecto a la formación en la Escuela en general, mi análisis no es exclusivo al DCC. En mi primera prueba de la Escuela me saqué un rojo, esencialmente porque la prueba no tenía nada que ver con lo que me habían enseñado. Esto es, lo que me preguntaron no se parecía a ninguno de los ejercicios que había hecho antes. Sin embargo, con el tiempo uno termina dándose cuenta que eso está bien, es normal, a ti te en-

señan las cosas de una manera pero las tienes que usar de otra. Y ese es un sello muy profundo en la Escuela, el cual ha marcado muchísimo mi ejercicio como emprendedor e innovador. Siento que el DCC entrena a sus estudiantes en este proceso. En aprender a adaptarse. En usar el conocimiento y no repetir simplemente "pares ordenados" de preguntas y respuestas.

Y en tu opinión, ¿fuiste una anomalía o eres representativo de un estudiante tipo en el DCC? ¿Ser innovador es algo que se puede enseñar o es simplemente una característica de ciertos estudiantes que solo se puede alimentar y quizás potenciar?

Creo que el DCC no es una escuela de negocios. Entonces, como emprendedor, es normal y está bien que sea una anomalía. Pero lo que el DCC sí hace, y su sello está en muchos exalumnos de mi generación, es forjar la capacidad de crear y pensar cosas de una manera distinta, lo cual puedes realizar en grupos chicos o en grupos grandes. No creo que el DCC tenga que ponerse la meta de formar muchos emprendedores. El conocimiento que te entrega se puede aprovechar en el emprendimiento como también en muchos otros ámbitos.

¿Qué iniciativa puede implementar un departamento para fomentar que salgan más de estos profesionales que "eligen creer" que se la pueden, que pueden tomar ideas y tecnolo-

gía "de las nubes" y bajarlas a tierra para generar un emprendimiento exitoso?

Soy muy fan del DCC así que no creo que deba tener un giro en 180 grados, pues ya lo está haciendo muy bien. Sin embargo, algo que no vi mucho en mi formación (no sé si habrá cambiado ahora) es enseñarle a la gente a fallar, a permitirse fallar. Entender que en el error y en los fracasos hay un aprendizaje que nunca vas a tener en los éxitos. Estos últimos solo refuerzan lo que tú creíste. Pero al fallar tienes una posibilidad de aprendizaje gigante, y quedas mucho mejor cuando lo sabes digerir y cuando hay un ecosistema que permite la falla. Creo que eso es un elemento muy importante a cultivar. Lo que haces con respecto a tus fracasos determina muchísimo lo que vas a poder hacer después, especialmente como emprendedor.

Cambiando de tema, tus emprendimientos fueron tecnológicos. ¿Podrías hablar un poco de cómo armaste tus equipos en ellos, en particular, de cómo usaste ingeniería de software para lograr producir tus objetivos? ¿Cómo es en Chile hoy emprender a partir de hacer software?

Hoy es muchísimo más fácil que hace diez años. El ecosistema ha mejorado una brutalidad estos últimos años. Cuando comencé Acepta.com las posibilidades de financiar una empresa de esas características eran bajas. Era todo absurdo, cuan-

do nos juntábamos los pocos emprendedores en esa época comentábamos cosas como "llevo cinco meses emprendiendo y planeo seguir haciéndolo". ¡Parecía una reunión de Alcohólicos Anónimos! Hoy día hay grupos grandes de emprendedores. Todo lo que se ha armado en Start-Up Chile, todo el financiamiento de capitales semilla, de las distintas incubadoras en universidades es significativo.

Hoy en día, en etapas tempranas de tu emprendimiento hay muchos fondos a los cuales puedes acceder. Levantar fondos de entre 40 a 60 millones es algo que logran no menos de mil emprendimientos al año. En mi época, no eran más de diez. Partir hoy es extraordinariamente más fácil. Sin embargo, continuar es muy difícil. De los mil emprendimientos que levantaron capital e intentaron algo seriamente el año pasado, no más de unos cincuenta tienen evidencia que su negocio puede ser bueno y continuar. Para esos cincuenta financiarse es muy difícil. Corfo ha hecho cosas interesantes como el capital semilla "expansión", liberado en 2014, pero lamentablemente los montos que se necesitan son mucho más grandes que los provistos por Corfo. Además, hay un problema no muy fácil de ver para un "computín", pero que mi lado "industrial" me lo permite detectar: en Chile, las rentabilidades en industrias como las inmobiliarias, mineras y similares, de muy bajo riesgo, son muy altas. Entonces, la rentabilidad versus riesgo de nuevas industrias (como la tecnológica) hace que no tenga mucho sentido in-



IMAGEN 1.
EN 2013 KIPU GANÓ EL DEMO DAY DE START UP CHILE.



IMAGEN 2.

KHIPU EN LAS REDES SOCIALES.

vertir en capital de riesgo. Es un asunto meramente económico. Si uno acepta que un inversionista quiere maximizar su rentabilidad minimizando el riesgo, no es claro que sea buena idea invertir en capital de riesgo en Chile. Ciento, aquí hay dos cosas: hay reglas del juego en Chile que hacen demasiado más fácil enriquecerse en industrias como las inmobiliarias, y por ende hay algo que corregir allí, lo cual se está discutiendo hoy en día. Pero por otro lado, se requieren políticas públicas mucho más fuertes para que haya disposición a, y hagan más fácil, invertir en emprendimientos tecnológicos. Hay rentabilidades muy altas en áreas de la economía que no favorecen al país.

Desde un punto de vista más técnico, ¿cómo fue esto de armar un emprendimiento donde el valor estaba dado por un software, por sobre quizás el grupo de gente que desarrolló ese software?

Esto fue fundamental en Khipu. Allí logramos armar un equipo humano cuyos méritos van mucho más allá de mi aporte. Este grupo implementa buena parte de las metodologías de integración continua, incluyendo pruebas unitarias, pruebas de integración, y pasos a producción semiautomatizados. En Khipu podemos pasar a producción nuevas versiones dos veces al día en forma bastante segura. Nosotros gastamos el 50% de nuestro esfuerzo en producir código y el otro 50% en que éste viva en un entorno que nos permita confiar en dicho código. Por lo tanto, tenemos

mucho velocidad para desarrollar, para adaptarnos, para incorporar características nuevas al software. Esto es fundamental pues somos un equipo muy chico. Nuestro competidor más chico tiene diez veces más ingenieros en desarrollo que nosotros. Un competidor grande cien veces.

Mirando números, nuestro presupuesto de comercialización llega a 300 millones de pesos. Transbank, en lo mismo, va a gastar 20 millones de dólares. En otras áreas se tiene algo similar: diferencias en presupuesto de 1 a 200. Por lo tanto, para competir, tienes que lograr productividades que hagan una diferencia gigante. Y eso no es algo que te lo de una persona genial, sino que depende de las metodologías. Notemos que las metodologías de integración continua son prácticamente imposibles de implementar en una empresa que ya está andando. Así, si partes desde cero una empresa con integración continua y mantienes esa capacidad, entonces tienes una ventaja sustentable en el largo plazo. Eso es una ventaja estratégica para nosotros.

¿Cómo ha sido tu experiencia en términos técnicos de este proceso de desarrollar tecnología? Porque tú vendes algo poco frecuente en un país como Chile.

Aquí vale la pena contrastar mi experiencia en armar Acepta.com versus armar Khipu. Son dos empresas muy distintas. La tecnología de Acepta.com era muy innovadora en su tiempo y, por lo tanto, vender esa tecnología era muy difícil. La tecnología de Khipu es muy innovadora en su diseño, en su concepto, y "hacia adentro", pero el problema que resuelve afuera, la solución que uno le ofrece a los clientes es algo que todo el mundo entiende y la verdad es que es bien fácil de vender. Efectivamente, luego de decir "te ofrezco una solución que te va a permitir cobrar por más canales, que te van a pagar más personas, que te va a costar menos dinero y va a ser más estable" ya estás listo, ¡has terminado de vender! Ahora, producirlo y financiar su producción, es mucho más complejo porque el inversionista tiene muchas más dificultades para entender que se puede producir algo tanto mejor que lo que ya existe para resolver el mismo problema. Y ahí sí es importante entrar en los detalles técnicos de por qué simplificar transferencias permite llegar a más personas que los pagos con tarjetas de crédito, permite entrar a una industria que todos creen es de los bancos pero que, en realidad, no lo es. Y es ahí donde vas abriendo muchos caminos que parecían cerrados.

IMAGEN 3.
EL TERMINAL DE PAGOS KHIPU.

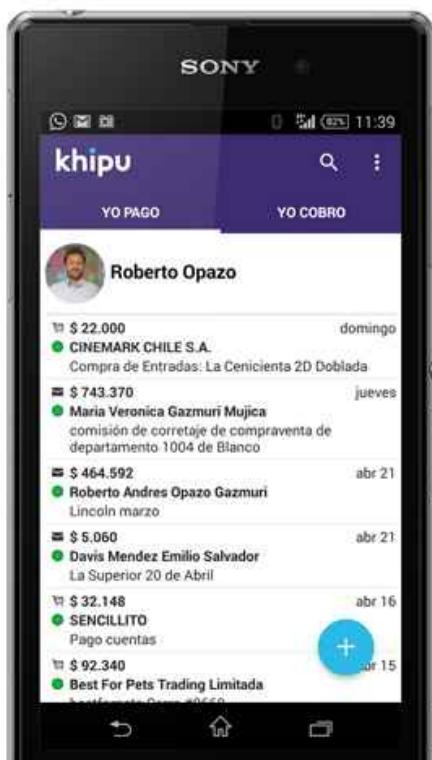


IMAGEN 4.
SECCIÓN “YO PAGO” EN LA APP DE KHIPU.



IMAGEN 5.
SECCIÓN “YO COBRO” EN LA APP DE KHIPU.

Se habla mucho que para un país pequeño como Chile lo mejor es diversificarse e invertir en tecnología. ¿Tiene sentido emprender en Chile trayendo tecnología de un nivel teórico a un nivel práctico, hacer lo que antiguamente se llamaba “transferencia tecnológica”?

Tengo un problema con el término “transferencia tecnológica”. Creo que debiera dársele otro nombre a la misma actividad. Ella consiste en que universidades y el mundo académico colaboren con empresas para producir innovación. “Colaboración para innovación” me parece un mucho mejor término. Cuando se habla de “transferencia tecnológica” queda la impresión que el conocimiento está en un lado y simplemente lo vas a transferir al otro. Suena infinitamente soberbio. Y a las empresas les carga, pues de partida miran en menos al mundo académico (el típico argumento de “quien enseña no tiene experiencia” o no sabe lo que realmente está pasando). Así que esta idea de que el conocimiento está en el mundo académico y se transfiere a las empresas les cae como “*patá en la guata*” a estas últimas. Como concepto de marketing, nació muerto. Pero sí creo en la innovación. Y sí creo en la colaboración entre el mundo académico y las empresas para la innovación.

¿Y cómo crees que esa colaboración puede concretarse mejor en un país como Chile, donde los tiempos e incentivos que tienen las empresas y el mundo académico son distintos?

La clave se llama “osmosis” (se ríe). ¿Cuáles son las fuentes de osmosis? Una es cuando alguien del mundo académico forma parte de un equipo de trabajo en una empresa, por ejemplo, cuando hay un grupo de estudiantes que van a ayudar en ciertos temas. Otra forma de osmosis es cuando gente de las empresas hacen docencia, como un profesor part-time que va a hacer clases a una universidad. Se debe buscar osmosis en ambas formas, para así finalmente conformar equipos de trabajo que estén en ambos lugares. El problema es que en esos equipos de trabajo la discusión siempre está en quién es el dueño de lo que se produzca. Ese es un problema natural entre las empresas y la universidad. Ambos quieren que-

darse con una parte, al menos, de la propiedad de lo que se produzca. Creo que las universidades no deberían aspirar más que al mérito. Y las empresas debieran quedarse con la propiedad. Simplemente porque lo otro no funciona.

También la “rotación” (profesores que temporalmente migran a la empresa privada y luego retornan al mundo académico) es una forma de osmosis que el tipo de evaluación docente actual no favorece. Un académico que decide hacer eso está perdido, muere académicamente. Desde la mirada de la empresa no es tan malo.

Creo que en general la visión que las empresas tienen de los académicos es equivocada. Creo que los académicos saben muchas cosas que son útiles para las empresas y, a su vez, las empresas no los aprovechan. El respeto por los académicos no está inculcado en la cultura de las empresas. De hecho, ¡ese respeto los estudiantes lo pierden cuando se titulan! Mientras uno es estudiante, valora mucho a sus profesores, pero cuando se titula y entra a una empresa, lo primero que te dicen es que “los profesores no saben nada” y “ahora vas a aprender” (se ríe).

En otras latitudes es común que un grupo de estudiantes en un rato de ocio inventen algo que transformó al mundo. Y en términos de calidad de estudiantes, no creo que los nuestros tengan algo que enviarles a los de ese tipo de instituciones. Sin embargo, ese tipo de creaciones no surge tan seguido acá. ¿Hay algo que quizás podríamos hacer mejor, convencerlos más que innovar es posible?

Primero que nada, cada vez se innova más en Chile. Y la gente cada vez se atreve más. No lo veo tan crítico como “allá sí y acá no”. Siento que estamos en un momento de transición, donde antes ocurría allá y acá no, pero ahora ocurre allá y acá poco, pero cada vez ocurre más. Y vamos en esa dirección, siento que todo el mundo quiere empujar en esa dirección. No hay mucha duda. Estamos descubriendo, experimentando cómo hacerlo. Ese es un proceso en marcha. Y como tal, de repente hay una tendencia a buscar leyes. Por ejemplo, “fail fast”: si tienes una idea, haz la



maqueta lo más rápido posible para salir al mercado, pruébala, y si falla, ningún problema, te vas a la próxima. Para algunas ideas eso puede ser una buena práctica, pero para la mayoría encuentro que no. Uno tiene que tener un poquito más de convicción, probar con más prudencia, empujar con más ganas. Apurarse en el fracaso no es lo mismo que no tenerle miedo al fracaso y aprender de la derrota. Hay otras ideas que efectivamente son bastante simples, y la única duda es si tienen mercado. Y allí es correcto: empaquetar rápido una muestra y ver cómo la recibe la gente.

Ahora bien, cuando uno quiere ser el primer país del mundo en usar firma electrónica para darle valor probatorio a los documentos tributarios, ese proyecto no es posible de realizar “fallando rápido”. Tienes que cambiar la ley, tienes que convencer a mucha gente, a muchas empresas. Yo creo que uno no debe estar demasiado desesperado por buscar las leyes universales del emprendimiento y de la innovación. Uno debe entender que ese mundo es tan diverso que en algunos casos aplican unas reglas y en otros casos, aplican otras.

¿Algún comentario final?

Políticas públicas. Como país, como comunidad, tenemos que subirnos todos al carro de las políticas públicas. Ellas no son algo que decidan los gobiernos de turno, se determinan por la academia, las empresas, la ciudadanía, a través de la prensa, en redes sociales. Entonces, empujar y empezar a pensar en términos de cuál es la política pública necesaria para resolver los distintos problemas que enfrentamos. Como individuos o como universidades quizás no somos lo suficientemente grandes como para producir grandes cambios, y por ello es necesario definir la política pública que va a empujar a esos grandes cambios.

Un ejemplo de política pública sería que las universidades no tengan la propiedad intelectual o industrial de las cosas que inventan sus académicos, o de las cosas que en colaboración con las empresas se producen. Eso facilitaría el tener negocios con empresas. Y es muy simple, una regla no más. Otra sería que en la evaluación docente se reconozca la experiencia que las per-

sonas tienen en la empresa. Es muy simple pero hay que ponerse de acuerdo en aceptar esa regla.

¿Cuál política crees que fomentaría la innovación?

El financiamiento hoy día es fundamental. Se ha hecho mucho en la capa inicial, pero ahora hay mucho por hacer en la segunda o tercera fase del ciclo de vida de un emprendimiento. Otra está asociada a qué enseñarle a los emprendedores. Los emprendedores siempre están orientados a desarrollar su producto, tienen una cierta fascinación por su producto. Pero para que su emprendimiento tenga éxito, necesitan gastar mucho tiempo en lograr resultados a través de otros, en gestionar gente, en financiar su emprendimiento, que no tiene que ver con abordar directamente el producto sino indirectamente: darle recursos humanos que lo produzcan y dar financiamiento para tener insumos. Y eso tiende a ser “una lata” pero probablemente es lo más determinante en el éxito de un emprendimiento. ■



IMAGEN 6.
PARTE DEL “EKHIPU” O EQUIPO DE KHIPU.

PROPUESTAS DE BUENAS PRÁCTICAS DE DESARROLLO DE SOFTWARE PARA PEQUEÑAS EMPRESAS DE SOFTWARE

El éxito de un proyecto es vital y crítico para la supervivencia de las Pequeñas Empresas de Desarrollo de Software (PEDS). Usualmente estas empresas no tienen una buena liquidez. No tienen mucho respaldo financiero. Muchas veces el fracaso de un solo proyecto puede significar la bancarrota total de estas empresas. Son muy frágiles. Por lo tanto, necesitan tener éxito en sus proyectos. Necesitan construir productos de software que satisfagan las necesidades reales de sus clientes dentro de los tiempos y presupuestos planificados. Uno de los factores que contribuye positivamente en dicho éxito es el uso de buenas prácticas de desarrollo de software. Por este motivo, existen diversas propuestas donde las PEDS pueden recurrir en busca de buenas prácticas que les ayuden a mejorar sus tasas de éxito.





ALCIDES QUISPE

Profesor Asistente Departamento de Ingeniería Informática, Universidad de Santiago de Chile. Doctor en Ciencias mención Computación, Universidad de Chile; Ingeniero de Sistemas, Universidad Católica de Santa María, Perú. Líneas de investigación: Ingeniería de Software - ingeniería de requisitos, procesos de software, e investigación empírica en ingeniería de software.

alcides.quispe@usach.cl



Actualmente nadie puede negar el importante rol que vienen jugando las PEDS dentro de la industria del software. Diversos estudios muestran que este tipo de empresas representan una amplia y creciente parte de la industria del software. En muchos casos son una pieza fundamental del crecimiento económico de un país [9, 27]. Sin embargo muchas de ellas tienen serios problemas con el éxito de sus proyectos. Terminan fuera de los plazos, excedidos en el presupuesto o las soluciones que entregan no satisfacen las necesidades reales del cliente [13].

¿QUÉ ES UNA PEDS?

Definir qué es una PEDS no es tarea fácil. Actualmente no hay un consenso sobre una única definición. Algunos estudios consideran como PEDS a aquellas empresas con menos de 100 empleados [32]. Otros las consideran como aquellas empresas con menos de 50 empleados [8]. Y otros las consideran como aquellas que tienen menos de 20 empleados [21, 33]. Otro criterio usado para definir una PEDS es el monto de facturación; sin embargo el número de empleados pareciera ser el más usado [3].

En Chile, según los resultados del diagnóstico de la industria TI desarrollado en 2003 [35], una PEDS es definida como aquella empresa que tiene entre 11 y 50 empleados. Según esta definición, este estudio muestra que en Chile al menos el 44% de las empresas pueden ser con-

sideradas como PEDS. Tomando en consideración esta misma definición, podemos ver que en Brasil las PEDS representan al menos un 50% [1]; en Canadá ellas son al menos el 28% [19], y en Finlandia ellas son al menos 23% [28]. Estos datos muestran que las PEDS representan un porcentaje no menor dentro de la industria de software.

CARACTERÍSTICAS DE LAS PEDS

Las PEDS tienen características que las distinguen completamente de las medianas y grandes empresas. Estas características hacen que cualquier solución propuesta para las medianas o grandes compañías no sea adecuada para este tipo de empresas. Por lo tanto, cualquier esfuerzo en proveer soluciones para éste tipo de empresas debería considerar las características de ellas.

Tipo de proyecto: las PEDS desarrollan proyectos que usualmente duran hasta seis meses [31, 24]. Sus proyectos están enfocados hacia nichos de negocio específicos debido a que se van especializando en ciertos dominios de negocio [2].

Tamaño del equipo de desarrollo: los equipos de desarrollo usualmente son pequeños. Varían entre tres a diez personas [29, 2]. El beneficio de usar equipos de desarrollo pequeños es que el costo de la coordinación se reduce considerablemente [17].

Recursos: las PEDS usualmente tienen acceso limitado a recursos humanos, tecnológicos y económicos [27, 5]. Los recursos económicos son considerados los más críticos para este tipo de empresas. Su liquidez depende mucho de las ganancias de sus proyectos [18, 20].

Personal de trabajo: Su personal tiene un nivel de preparación medio a bajo. Las PEDS no siempre pueden contratar personal altamente calificado y experimentado [22]. Usualmente contratan profesionales con un *know-how* general en vez de especialistas [7]. Su personal, principalmente aquellos dedicados al desarrollo de software, frecuentemente trabajan en diferentes roles en los proyectos [24, 17].

Procesos de software: el proceso de desarrollo de software de las PEDS es típicamente informal y muchas veces es un proceso *ad-hoc* [12, 36].

Administración de proyectos: las PEDS planean, organizan, dirigen, monitorean y controlan sus proyectos con un alto nivel de informalidad [4, 23].

Complejidad de sus productos: los proyectos de software ejecutados por las PEDS usualmente son para construir productos de software simples

[12]. En consecuencia, estas empresas pueden abordar estos proyectos con procesos simples.

Estructura organizacional: comúnmente las PEDS tienen una estructura organizacional informal con roles y responsabilidades vagamente definidos [24, 23, 17]. La estructura organizacional es casi siempre plana o tiene pocos niveles de jerarquía [7, 6].

Existen muchos factores que influyen en el éxito de los proyectos de software. Sin embargo, es bien reconocido que el uso de buenas prácticas de desarrollo de software contribuye positivamente al éxito de los proyectos. Por lo tanto, el uso de estas prácticas debería ser motivo de atención por parte de las PEDS.

Varios países y algunas organizaciones han hecho esfuerzos con el objetivo de proponer un conjunto de buenas prácticas de desarrollo de software para las pequeñas empresas de software (**ver Tabla 1**). Estas propuestas fueron diseñadas tomando en consideración las características de las PEDS.

PROPUESTAS DE BUENAS PRÁCTICAS DE DESARROLLO DE SOFTWARE PARA LAS PEDS

Como fue mencionado en la sección anterior, una de las características más críticas de las PEDS es su ajustada liquidez. No tienen un fuerte respaldo financiero. Para muchas de ellas el fracaso de un solo proyecto podría significar la bancarrota total de la empresa. Son empresas frágiles. Por lo tanto, necesitan tener éxito en sus proyectos para seguir siendo parte activa de la industria del software.

ISO/IEC 29110 (BASIC PROFILE)

El standard ISO/IEC 29110 (Basic Profile) [15] fue diseñado exclusivamente para empresas de desarrollo de software con menos de 25 empleados. Fue desarrollado por la Organización Internacional para la Normalización (ISO) en conjunto con la Comisión Electrónica Internacional (IEC). El objetivo de este nuevo estándar es que las

ID	PROPIEDAD	TIPO	AÑO	PÚBLICO OBJETIVO	BASADO EN	DD ¹	IDIOMA ²	# AP ³
1	ISO/IEC 29110 (Basic Profile) [15]	Estándar de Software	2011	Todos los países	MoproSoft, ISO/IEC 12207, ISO/IEC 15504, ISO 9000	Si	Español	2
2	Tutelkan [34] de Referencia	Proceso de Software	2009	Chile	CMMI-Dev, ISO 9001, ISO/IEC 15504	Si	Español	18
3	Competisoft [26] de Referencia	Proceso de Software	2007	Países Iberoamérica	MoproSoft, MPS.BR, Métrica, Mantema, AgileSPI	Si	Español	10
4	MPS.BR [37]	Estándar de Software	2003	Brasil	ISO/IEC 12207, ISO/IEC 15504, CMMI-DEV	Si	Español	19
5	MoproSoft [25]	Estándar de Software	2002	México	ISO/IEC 12207, ISO/IEC, ISO/IEC 15504, ISO 9000, CMMI, PMBOK	No	Español	9

TABLA 1.
PROPUESTAS DE PRÁCTICAS PARA PEDS.

1 DD : documentación disponible desde Internet.

2 Idioma : idioma en la que está redactada la documentación.

3 #AP : número de áreas de proceso.

pequeñas empresas de software puedan mejorar el desempeño de sus procesos y de esta manera logren un reconocimiento internacional dentro de la industria del software. Propone que una PEDS debería implementar dos áreas de proceso, Gestión del Proyecto e Implementación del Software. Para cada una de estas áreas de proceso, el estándar 29110 provee un conjunto detallado de buenas prácticas [16].

TUTELKAN

El proyecto Tutelkan [34] propone una estrategia de mejoramiento de procesos de software para las pequeñas empresas de desarrollo de software de Chile. Fue el resultado de un proyecto financiado por el Gobierno chileno. En dicho proyecto participaron dos universidades chilenas (Universidad Técnica Federico Santa María y la Universidad de Chile), la Sociedad Chilena de Software y Servicios (GECHS), la Asociación Chilena de Empresas de Tecnología de Información (ACTI), el Software Process Improvement Network (SPIN Chile) y la empresa Kepler Technology S.A. El objetivo de este proyecto fue desarrollar un proceso de software de referencia de acceso público, el cual podría ser usado por las pequeñas empresas de software chilenas para definir y documentar sus propios procesos de desarrollo de software que les permitiera lograr una futura certificación ISO 9001 y CMMI. Esta propuesta presenta una serie de buenas prácticas de desarrollo de software para cada una de las 18 áreas de proceso que propone [10, 11] (ver Tabla 2).

COMPETISOFT

Competisoft es un marco metodológico orientado a mejorar los procesos de las pequeñas y medianas empresas de desarrollo de software de Iberoamérica [26]. Su desarrollo fue financiado por el Programa Iberoamericano de Ciencia y Tecnología para el Desarrollo (CYTED). Fue el resultado de un trabajo conjunto de investigadores provenientes de más de diez universidades iberoamericanas, el Instituto

CATEGORÍA	ÁREAS DE PROCESO
Gestión de Proyectos	<ul style="list-style-type: none"> Administración de Requerimientos Planificación del Proyecto Monitoreo y Control del Proyecto Administración de Acuerdos con Proveedores Medición y Análisis Administración de Calidad de Procesos y Productos Administración de Configuraciones
Desarrollo y Mantención de Software	<ul style="list-style-type: none"> Desarrollo de Requerimientos Análisis y Diseño Programación Pruebas Instalación
Gestión Avanzada de Proyectos	<ul style="list-style-type: none"> Gestión de Riesgos Evaluación Formal de Decisiones Administración Integrada del Proyecto
Gestión de Procesos	<ul style="list-style-type: none"> Definición de Procesos Organizacionales Mejoramiento de Procesos Organizacionales Entrenamiento Organizacional

TABLA 2.
ÁREAS DE PROCESO DE TUTELKAN.

CATEGORÍA	ÁREAS DE PROCESO
Alta Dirección	<ul style="list-style-type: none"> Gestión de Negocio
Gerencia	<ul style="list-style-type: none"> Gestión de Procesos Gestión de Cartera de Proyectos Gestión de Recursos Gestión de Recursos Humanos Gestión de Bienes, Servicios e Infraestructura Gestión de Conocimiento
Operación	<ul style="list-style-type: none"> Administración del Proyecto Desarrollo de Software Mantenimiento de Software

TABLA 3.
ÁREAS DE PROCESO DE COMPETISOFT.

Argentino de Normalización y Certificación (IRAM), y pequeñas empresas de desarrollo de software de ocho países de Iberoamérica. Competisoft incluye buenas prácticas de desarrollo

de software para cada una de las diez áreas de procesos que propone [14] (ver Tabla 3).

MPS.BR

MPS.BR es un programa de mejoramiento de procesos de desarrollo de software para las pequeñas y medianas empresas de la industria de software brasileña [37]. Este programa es coordinado por la Asociación para Promoción de la Excelencia del Software Brasileño (SOFTEX). Cuenta con el apoyo del Ministerio de Ciencia, Tecnología e Innovación (MCTI), del Servicio Brasileño de Apoyo a las Micro y Pequeñas Empresas (SEBRAE) y del Banco Interamericano de Desarrollo (BID/FUMIN). Esta norma describe en detalle un conjunto de buenas prácticas para cada uno de las 19 áreas de procesos que propone [30] (**ver Tabla 4**).



• Gestión de Proyectos	• Gestión de Portafolios de Proyectos	• Gestión de Reutilización	• Verificación
• Gestión de Requisitos	• Medición	• Desarrollo de Requisitos	• Desarrollo para Reutilización
• Adquisición	• Evaluación y Mejora del Proceso Organizacional	• Integración del Producto	• Gestión de Decisiones
• Gestión de Configuración	• Definición del Proceso Organizacional	• Diseño y Construcción del Producto	• Gestión de Riesgos
• Aseguramiento de la Calidad	• Gestión de Recursos Humanos	• Validación	

TABLA 4.
ÁREAS DE PROCESO DE MPS.BR.

MOPROSOFT

Moprossoft [25] es un standard mexicano creado para ayudar a las pequeñas empresas de desarrollo de software mexicanas a mejorar sus procesos de software. Fue desarrollado a solicitud de la Secretaría de Economía de México con el apoyo de la Facultad de Ciencias de la Universidad Nacional Autónoma de México (UNAM). Moprossoft incluye una serie de prácticas para cada una de sus nueve áreas de procesos que propone (**ver Tabla 5**).

De las cinco propuestas descritas anteriormente se puede observar que Tutelkan y MPS.BR tienen casi el mismo número de áreas de proceso (18 y 19 respectivamente). Lo mismo ocurre con Moprossoft y Competisoft (9 y 10 respectivamente). Solamente el estándar ISO/IEC 29110 propone un número muy reducido de áreas de proceso (2 procesos). Estos resultados sugieren que quizás la mejor opción, para aquellas PEDS que desean

mejorar sus procesos de software, es iniciar con el standard ISO/IEC 29110. Incluye un número minimal de buenas prácticas.

Todas las propuestas, con excepción de Moprossoft, proporcionan acceso libre a su documentación. Dichas propuestas pueden ser bajadas (sin costo alguno) desde sus sitios web y usadas

directamente. En consecuencia, no se requiere que las PEDS tengan que invertir dinero en su adquisición.

También se puede observar que todas ellas están redactadas en idioma español. En consecuencia, las PEDS no tendrían problemas para leerlas y comprenderlas.

CATEGORÍA	ÁREAS DE PROCESO
Alta dirección	<ul style="list-style-type: none"> • Gestión de Negocio
Gerencia	<ul style="list-style-type: none"> • Gestión de Procesos • Gestión de Proyectos • Gestión de Recursos - Recursos Humanos y Ambiente de Trabajo - Bienes Servicios e Infraestructura - Conocimiento de la Organización
Operación	<ul style="list-style-type: none"> • Administración de Proyectos Específicos • Desarrollo y Mantenimiento de Software

TABLA 5.
ÁREAS DE PROCESO DE MOPROSOFT

CONCLUSIONES

LAS CARACTERÍSTICAS PARTICULARES DE LAS PEQUEÑAS EMPRESAS DE DESARROLLO DE SOFTWARE (PEDS) MUESTRAN QUE ESTE TIPO DE COMPAÑÍAS SON TOTALMENTE DISTINTAS DE LAS MEDIANAS Y GRANDES COMPAÑÍAS. POR LO TANTO, ESTAS CARACTERÍSTICAS DEBERÍAN SER TOMADAS EN CONSIDERACIÓN AL MOMENTO DE DISEÑAR SOLUCIONES PARA ESTE TIPO DE EMPRESAS.

ACTUALMENTE EXISTEN PROPUESTAS DE PRÁCTICAS DE DESARROLLO DE SOFTWARE QUE PUEDEN AYUDAR A ESTAS EMPRESAS A MEJORAR LA FORMA EN QUE DESARROLLAN SUS PRODUCTOS Y DE ESTA MANERA INCREMENTAR SU TASA DE ÉXITO. LA VENTAJA DE ESTAS PROPUESTAS ES QUE HAN SIDO DESARROLLADAS TOMANDO EN CONSIDERACIÓN LAS CARACTERÍSTICAS DE ESTAS COMPAÑÍAS. POR LO TANTO, LAS PEDS PUEDEN RECURRIR A ELLAS CON LA CERTEZA DE QUE ESTARÁN USANDO SOLUCIONES QUE CALZAN CON LAS CARACTERÍSTICAS QUE LAS HACEN DISTINTAS DE LAS MEDIANAS Y GRANDES COMPAÑÍAS. ■

REFERENCIAS

- [1] ABES Software. "Brazilian Software Market Scenario and Trends", 2014.
- [2] J. Aranda, S. Easterbrook, G. Wilson. "Requirements in the wild: How small companies do it". Proc. of the 15th IEEE Requirements Engineering Conference, pp. 39–48, 2007.
- [3] M. Ayyagari, T. Beck, A. Demirguc-Kunt. "Small and medium enterprises across the globe". Journal of Small Business Economics, Springer, Vol. 29, No. 4, pp. 415–434, 2007.
- [4] J. Batista, A.D. De Figueiredo. "SPI in a Very Small Team: a Case with CMM". Journal of Software Process: Improvement and Practice, John Wiley & Sons, Ltd., Vol. 5, No. 4, pp. 243–250, 2000.
- [5] J. Dorr, S. Adam, M. Eisenbarth, M. Ehresmann. "Implementing Requirements Engineering Processes: Using Cooperative Self-Assessment and Improvement". IEEE Software, Vol. 25, No. 3, pp. 71–77, May/Jun. 2008.
- [6] K. El Emam. "Multi-Method Evaluation of the Practices of Small Software Projects". Proc. of the 1st Int. Research Workshop for Process Improvement in Small Settings, pp. 12–17, 2005.
- [7] L. Estrin, J. Foreman, S. Garcia. "Overcoming Barriers to Technology Adoption in Small Manufacturing Enterprises (SME)". Technical Report CMU/SEI-2003-TR-012, 2003.
- [8] European Commission. "The new SME definition User guide and model declaration". White Paper, May 2003. http://ec.europa.eu/enterprise/policies/sme/files/sme_definition/sme_user_guide_en.pdf (última visita: junio 2015).
- [9] M.E. Fayad, M. Laitinen, R.P. Ward. "Software Engineering in the Small". Communications of the ACM, Vol. 43, No 3, pp. 115–118, Mar. 2000.
- [10] Federico Santa María Technical University (Chile). "Tutelkan Reference Process (TRP) Versión 2.0 Parte 2: TRP Básico", 2009. <http://www.tutelkan.info/blog/wp-content/uploads/2009/06/trp-v20-basico.pdf> (última visita: junio 2015).
- [11] Federico Santa María Technical University (Chile). "Tutelkan Reference Process (TRP) Versión 2.0 Parte 3: TRP Avanzado", 2009. <http://www.tutelkan.info/blog/wp-content/uploads/2009/06/trp-v20-avanzado.pdf> (última visita: junio 2015).
- [12] M. Harris, K. Aebsicher, T. Klaus. "The whitewater process: software product development in small IT businesses". Communications of the ACM, Vol. 50, No. 5, pp. 89–93, 2007.
- [13] C. Hofer. "Software development in Austria: results of an empirical study among small and very small enterprises". Proc. of the 28th IEEE EUROMICRO Conference, pp. 361–366, 2002.
- [14] Ibero-American Program of Science and Technology (CYTED). "Competisoft Document-Process Improvement to Promote the Competitiveness of the Ibero-American Small and Me-

- dium Software Industry" (in Spanish), 2008. http://alarcos.esi.uclm.es/ipsw/doc/Competisoft-modelo_v1.pdf (última visita: junio 2015).
- [15] ISO/IEC. "ISO/IEC 29110 (Basic Profile)", 2011.<http://profs.etsmtl.ca/claporte/English/VSE/index.html> (última visita: junio 2015).
- [16] ISO/IEC. DTR 29110-5-1-2 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) - Part 5: Management and Engineering Guide-Basic VSE Profile, 2011. http://standards.iso.org/ittf/PubliclyAvailableStandards/c051153_ISO_IEC_TR_29110-5_1_2011.zip (última visita: junio 2015).
- [17] S. Jantunen. "Exploring software engineering practices in small and medium-sized organizations". Proc. of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering, ACM, pp. 96–101, 2010.
- [18] B. Keane, I. Richardson. "Quality: Attitudes and Experience Within the Irish Software Industry". LNCS Software Process Improvement, Springer, Vol. 3792, pp. 49–58, 2005.
- [19] C.Y. Laporte, S. Alexandre, A. Renault. "Developing International Standards for Very Small Enterprises" IEEE Computer, Vol. 41, No. 3, pp. 98–101, Mar. 2008.
- [20] C.Y. Laporte, S. Alexandre, R.V. O'Connor., "A software engineering lifecycle standard for very small enterprises", Proc. of the 15th European System & Software Process Improvement and Innovation, Springer, Vol. 16, pp. 129–141, 2008.
- [21] R. Matulevicius. "Survey of Requirements Engineering Practice in Lithuanian Software Development Companies". Information Systems Development, Springer, pp. 327–339, 2005.
- [22] O. Mondragon. "Addressing infrastructure issues in very small settings". Proc. of the 1st Int. Research Workshop for Process Improvement in Small Settings, pp. 5–10, 2005.
- [23] J. Montilva, J. Barrios, M. Rivero. "Training requirements and profiles for software engineers in Micro and Small Enterprises" (in Spanish). CLEI 2009.
- [24] S.F. Ochoa, J.A. Pino, L.A. Guerrero, C. Collazos. "SSP: A Simple Software Process for Small-Size Software Development Projects". Proc. of the IFIP International Workshop on Advanced Software Engineering, Springer Science + Business Media, Vol. 219, pp. 94–107, 2006.
- [25] H. Oktaba. "MoProSoft: A Software Process Model for Small Enterprises". Proc. of the 1st Int. Research Workshop for Process Improvement in Small Settings. Special Report CMU/SEI-2006-SR-001.
- [26] H. Oktaba, F. García, M. Piattini, F. Ruiz, F.J. Pino, C. Alquicira. "Software process improvement: The Competisoft project". IEEE Computer, Vol. 40, No. 10, pp. 21–28, 2007.
- [27] I. Richardson, C.G. von Wangenheim. "Why Are Small Software Organizations Different?". IEEE Software, Vol. 24, No. 1, pp. 18–22, Jan./Feb. 2007.
- [28] M. Rönkkö, E. Eloranta, H. Mustaniemi, O. Mutanen, J. Kontio. "National Software Industry Survey 2008: The Finnish Software Industry in 2007". Helsinki University of Technology, 2007.
- [29] S. Rowe. "Project Management for Small Projects". Management Concepts, 2006.
- [30] Association to Promote Excellence in Brazilian Software (SOFTEX). "Modelo de Referencia MR-MPS Reference Model" (in Spanish), 2011. URL: <http://www.softex.br/mpsbr/guia-em-espanhol/> (última visita: junio 2015).
- [31] W. Stein. "A Web Software Process for Small or Medium-Sized Projects Focused on the Chilean Scenario". Engineering Thesis (In Spanish), Computer Science Department, University of Chile, 2003.
- [32] A. Tahir, R. Ahmad. "Requirements Engineering Practices - An Empirical Study". Proc. of the IEEE Int. Conf. on Computational Intelligence and Software Engineering, pp. 1–5, 2010.
- [33] A. Talbot, A. Connor. "Requirements Engineering Current Practice and Capability in Small and Medium Software Development Enterprises in New Zealand". Proc. of the 9th IEEE Int. Conf. on Software Engineering Research, Management and Applications, pp. 17–25, 2011.
- [34] G. Valdés, H. Astudillo, M. Visconti, C. López. "The Tutelkan SPI Framework for Small Settings: A Methodology Transfer Vehicle". Journal of Systems, Software and Services Process Improvement, Springer, Vol. 99, pp. 142–152, 2010.
- [35] M. Varela. "Diagnosis of the Chilean IT Industry – 2003. Diagnóstico de la industria de las tecnologías de la información en Chile 2003". (in Spanish). White Paper, CHILE INNOVA Program. 2003. http://www.dcc.uchile.cl/~aquispe/docs/tics_chile_2003.pdf (última visita: junio 2015).
- [36] C.G. von Wangenheim, T. Varkoi, C.F. Salviano. "Standard based software process assessments in small companies". Journal of Software Process: Improvement and Practice, John Wiley & Sons, Ltd., Vol. 11, No. 3, pp. 329–335, 2006.
- [37] K.C. Weber, E. Araújo, A. Rocha, C.A.F. Machado, D. Scalet, and C.F. Salviano. "Brazilian Software Process Reference Model and Assessment Method". LNCS Computer and Information Sciences, Springer, Vol. 3733, pp. 402–411, 2005.

MAGÍSTER EN
**TECNOLOGÍAS DE
LA INFORMACIÓN** (VESPERTINO)



EL PROGRAMA

El Programa ofrece especialización y perfeccionamiento para lograr una aplicación innovadora, segura y eficaz de las soluciones de TI en diversos ámbitos.

Busca formar especialistas de gran capacidad analítica y sólidas bases en aspectos teóricos y aplicados.

OBJETIVOS ESPECÍFICOS

- Especializar y profundizar conocimientos en áreas de interés informático, tales como gestión y desarrollo de proyectos en TI, ingeniería y calidad de software, ingeniería de datos y seguridad computacional.
- Fortalecer los aspectos transversales relacionados con el trabajo en equipo y liderazgo.
- Desarrollar individualmente un Proyecto de Grado aplicado en el ambiente de trabajo del estudiante.

ÁREAS DE DESARROLLO

Este Magíster presenta 5 áreas de desarrollo y especialización:

- Gestión de proyectos informáticos
- Ingeniería de software
- Calidad de software
- Ciencias e Ingeniería de Datos
- Seguridad Computacional

[www.dcc.uchile.cl/magister_ti]

CONSULTA TAMBIÉN POR NUESTROS DIPLOMAS DE POSTÍTULOS:



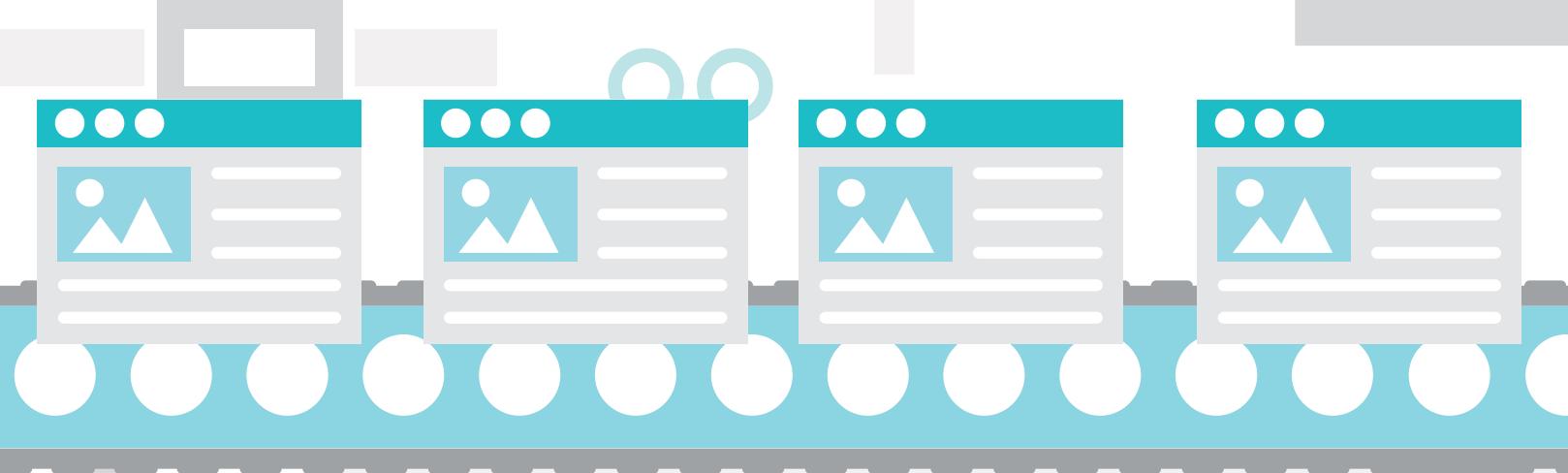
Educación Continua, una red de conocimientos

- Gestión Informática
- Ingeniería y Calidad de Software
- Seguridad Computacional
- Ciencias e Ingeniería de Datos
- Ingeniería de Software

www.dcc.uchile.cl/pec



010 001 101 10
00111001000110010
1110101011110101011
1010111000101011100
111100100 111 10
0101 11 101 1



www.dcc.uchile.cl
revista@dcc.uchile.cl