



# RESISTENCIA AL CAMBIO EN TORNO A LAS NTIC

Teoría General de Sistemas y Teoría de la Complejidad como claves de análisis del desempeño de docentes de Tecnología y encargados de Laboratorios de Informática en el nivel secundario de Mendoza

Analista en Sistemas Sergio Ariel Alonso

Dirigido por Prof. Dr. Jorge C. García-Carbajo

Mendoza – Noviembre de 2017





**Tesina: Resistencia al cambio en torno a las NTIC**

Teoría General de Sistemas y Teoría de la complejidad como claves de análisis del desempeño de docentes de Tecnología y encargados de Laboratorios de Informática en el nivel secundario de Mendoza

**Universidad Tecnológica Nacional – Facultad Regional Mendoza**

Licenciatura en Tecnología Educativa

Analista en Sistemas Sergio Ariel Alonso

Dirigido por Prof. Dr. Jorge C. García-Carbajo

Mendoza – noviembre de 2017

Diseño y realización de cubierta: Verónica Martínez

**Fotografía de base para tapa: Blankstock/Deposit photos**

## **Agradecimientos**

Deseo agradecer especialmente a mi director de Tesis, Prof. Jorge García Carbajo, quien llegando mucho más allá de su deber profesional, me acercó a la verdadera naturaleza del trabajo de investigación. De su inspiración por el detalle parte mi férreo deseo de crear nuevas y mejores producciones, tales que dignifiquen sus largas horas de minuciosa dedicación, en la creencia que ha visto algo de potencial en mis rudimentarios intentos de explicación. Gracias por su rigurosidad, por su inmenso conocimiento, y por su asombrosa capacidad para trascender campos. Gracias por convertirse en un maestro y en un guía dentro de un trabajo que dejó de ser un formalismo académico para convertirse en una aventura del conocimiento.

Agradezco asimismo la dedicación de mi ayudante de tesis, Sofía Cano, quien con su paciencia y especiales condiciones de psicóloga, profundizó en su parte de las entrevistas hasta obtener pistas que yo nunca hubiera logrado fuera del terreno teórico.

Agradezco profundamente a mi esposa, Verónica y a mi hija Guadalupe por apoyarme incondicionalmente en mis largas porfías, intentando ahondar en disciplinas que no eran las mías.

A mis padres, por haberme introducido tempranamente, con muchos esfuerzos tanto en tecnología como en el trabajo duro.

A mi madrina “Kiki” y a mis tías Enriqueta y Tita, quienes despejaron la oscuridad de mis primeros años a fuerza de cariño, de libros y de enigmáticos aforismos. Ellas pintaron de colores un mundo, para que otros lo vivieran.



## Índice de contenido

Capítulo 1: El perfil como fuente de resistencia.....	21
1.1. El ingresante autoengañado.....	21
1.1.1. El examen de ingreso.....	24
1.1.2. El perfil equivocado: la habilitación de cada título.....	25
1.1.3. La habilitación, ¿genera competencias para...?.....	34
1.2. El alumno desapasionado a mitad de carrera.....	40
1.2.1. El egresado lisiado.....	42
1.2.2. El egresado competente... en otros campos.....	44
1.3. El mito de la elección consciente del software, y sus consecuencias en los docentes de informática y tecnología.....	46
1.3.1. El software como una nueva zona de confort.....	47
1.3.2. El analista que no analiza, el ingeniero que no ingenia.....	52
1.3.3. Semblanza del egresado devenido en docente.....	53
Capítulo 2: Resistencia al cambio.....	55
2.1. Teorías asociadas al cambio.....	55
2.2. Adaptabilidad y cambio cognitivo: la plasticidad mental y la innovación.....	56
2.3. Deficiencias en el abordaje de la complejidad durante las carreras de sistemas. .	57
2.4. Mecánica en lugar de sistemas.....	64
Capítulo 3: El Lobo Solitario.....	69
3.1. Impacto de la resistencia al cambio en el docente:.....	72
3.1.1. Cese de investigación.....	72
3.1.2. Chauvinismo del idioma materno.....	76
3.1.2.1. Breve apología por el lenguaje universal <i>de facto</i> .....	77
3.1.3. El docente en la torre de marfil.....	83
3.1.4. En el país de los ciegos.....	85
3.1.5. El <i>failsafe</i> del incapaz.....	90
3.1.6. El bloqueo.....	92
3.1.7. El necio encubierto.....	102

3.1.8. El problema de la escuela vista como una empresa.....	104
3.2. Bloqueo entre los bloqueos.....	106
3.2.1. Introducción.....	106
3.2.2. El Nudo Gordiano.....	108
3.2.3. Pensamiento lateral creativo: el problema de crear la solución.....	110
3.2.4. Las listas de élite.....	112
3.2.5. Impacto sobre los alumnos.....	115
3.2.5.1. Neofobia por inducción.....	115
3.2.5.2. Consumidores y productores.....	116
Capítulo 4: Propuesta metodológica.....	125
4.1. Enfoque de Investigación.....	125
4.2. Método de Investigación.....	126
4.3. Población y tamaño de muestra.....	127
4.4. Instrumento de recolección de datos.....	129
4.4.1. Entrevistas en profundidad.....	129
4.4.2. Fichaje documental.....	129
4.5. Análisis de la Información: Planteo Teórico.....	131
Capítulo 5: Análisis de la información: personal de laboratorio, docentes de informática y tecnología en acción.....	133
Capítulo 6: Análisis de Datos: prospección de un modelo sistémico complejo de resistencia al cambio.....	187
Conclusión.....	201
Apéndice A - Entre <i>softwares</i> comerciales y liberados.....	215
A.1. Introducción.....	215
A.2. La superestructura <i>Wintel</i> de los <i>software</i> comerciales.....	216
A.3. Inducción a la obsolescencia programada: el dilema del gasto energético.....	220
A.3.1. Debates contemporáneos en torno al lugar y el impacto de las nuevas tecnologías.....	221
A.4. Prejuicios verdaderos y prejuicios falsos.....	226

A.4.1. Gratuidad y la calidad del <i>software</i> .....	227
A.4.2. Totalidad <i>versus</i> Modularidad.....	228
A.4.3. Protointerfases resistentes <i>ex professo</i> .....	229
A.4.4. El condicionamiento de las <i>interfases visuales</i> .....	232
Apéndice B: Propuestas de escenarios educativos para el docente de TIC.....	235
B.1. Introducción.....	235
B.2. El <i>hacking</i> bien entendido.....	236
B.3. Bienvenidos al tren.....	238
B.4. La <i>netbook</i> como laboratorio.....	239
B.5. La CLI como protolenguaje de programación.....	243
B.6. <i>Hardware</i> Libre.....	248
Apéndice C: El pensamiento estratégico necesario en las carreras tecnológicas.....	250
C.1. Introducción.....	250
C.2. Filtros en flujos de comunicación.....	256
C.2. Tratamiento terapéutico en docentes de tecnología para afrontar cambio y complejidad.....	260
Apéndice D: ¿Aprender a resolver problemas, o resolver los problemas de las empresas? .....	267
Apéndice E: Hurto de <i>software</i> en el aula.....	269
E.1. Principio económico de la escasez.....	269
E.2. Apología del hurto por parte del docente.....	270
E.3. Cleptoparasitismo.....	272
E.4. El <i>software</i> al rescate de las utopías.....	275
E.5. Abreviaturas empleadas.....	279
E.6. Resumen.....	284
Corolario: Hacia la extinción de Tecnología como asignatura.....	286
Bibliografía.....	288

## Introducción

Se ha determinado que no hay un uso de nuevas tecnologías, sino un consumo de computadoras entregadas por el gobierno, las cuales se utilizan como máquinas de escribir, calculadoras sofisticadas, para simple solaz de los alumnos o de los beneficiarios del plan Conectar Igualdad. La selección de *software* existente en las *netbooks*, que incluye programas educativos de toda índole, no ha sido usada en la materia de Tecnología más allá de la ofimática básica. Tampoco hay transferencia horizontal significativa hacia otras disciplinas. Esta percepción se corresponde con el uso que se le ha dado a las computadoras de escritorio compradas con esfuerzo antes del programa Conectar Igualdad. Además, teniendo la escuela y los profesores la potestad de introducir "nuevas" tecnologías a principio de año, la coincidencia general es mantener la brecha digital tal cual se encuentra.

Estos comportamientos de no-cambio, esta insistencia en mantener los contenidos de Tecnología de 1990 es cuando menos alarmante. En tanto que en otros espacios curriculares han habido varios cambios en los contenidos, fusiones, e inclusión de nuevos conceptos, tales como ecología, metodología de la investigación, historia de la ciencia, química ambiental, filosofía, economía social, formación ética y otros, en tecnología se enseña lo mismo que hace 20 años.

Estas observaciones podrían ser defendidas desde un punto de vista pragmático, de protección de los educandos ante la complejidad. Sin embargo ante propuestas de cambio se desencadenan situaciones de alta resistencia. El solo hecho de cambiar *Word* por *Openoffice*, o de *Windows* por *Linux* es motivo para llegar a extremos como abandonar completamente el uso de los equipos (cuando son equipos de escritorio, propios del establecimiento).

Para añadir complejidad a esta aseveración, ninguno de los siete docentes de Tecnología que constituyeron el estudio exploratorio del campo presentaron una justificación técnica o pedagógica a fin de legitimar sus posturas resistentes frente a las NTIC. Tampoco la opinión de uno prevaleció sobre la del resto. Los individuos observados no tenían prácticamente relación entre sí. Al ser consultados al respecto, encargados de laboratorio de otras escuelas admitieron que el fenómeno también era frecuente, lo cual sugiere un probable y sorprendente patrón de comportamiento en función de la representación social que tienen de sí mismos los profesores de Tecnología, y los propios encargados de laboratorio.

Por tal motivo, se presenta como objeto de estudio al docente y a su actitud frente a los cambios propuestos por las NTIC.

Una mirada superficial indicaría que los docentes de NTIC poseen capacidades superiores al promedio de la población para enfrentar la complejidad técnica. Para medir esta capacidad de impartir enseñanza en el terreno técnico, se puede evaluar por un lado en competencias en el terreno técnico, y por el otro como habilidades para hacer frente a situaciones nuevas. En el primer caso Dirección General de Escuelas confía solo en la institución educativa que ha librado al egresado al exterior. Es decir, no toma partido sobre estas habilidades. Para el segundo caso, dirime la situación contratando cada año la entidad que realiza los exámenes psicofísicos (OSEP, Ganun, aunque puede variar). Sin embargo algo en el proceso falla, por cuanto los informáticos imparten apenas algo más de lo que manejan los usuarios normales. Más adelante, durante las entrevistas, se podrá comprobar la inexistencia de actividades que se aparten del programa (Office de Microsoft y algo de web), ausencia de presentación a concursos u olimpiadas, conformación de proyectos o talleres de programación, robótica, diseño, etc. Considerando la fuerte presión a la que son sometidos estos

organismos por docentes que son rechazados<sup>1</sup>, prácticamente se puede aseverar que Dirección General de Escuelas queda a salvo del problema, mediante el mecanismo de trasladarlo: si el profesional es un incompetente, la culpa la tiene su universidad. Si no pasa el psicofísico, las acciones legales caerán sobre el médico o sobre el organismo de control.

Para no caer en subjetividades, no se continúa aquí la línea de investigación sobre este fenómeno, lo cual requeriría de otro conjunto de hipótesis y demostraciones. Pero se utiliza otra forma para demostrar como se encuentra instalado el problema de afrontar cambio y complejidad. La forma de ponerlo de manifiesto es el tratamiento que le otorgan los docentes de tecnología y de informática a *Linux*, el sistema operativo libre incluido en forma explícita en las *laptops* del proyecto Conectar Igualdad.

A pesar de las buenas intenciones de la comunidad de *software* libre, *Linux* parece ser el fusible que hace evidente la resistencia. Ante una cantidad conjunta de variaciones superior al promedio diario, es decir, más de dos o tres desafíos técnicos, tales como cambios de *interfase*, herramientas, accesorios, administrador de archivos, o aplicaciones, los mencionados sujetos de estudio lo sitúan como cambios estructurales a gran escala, con verdaderas antipatías y resistencias a su adopción, sin importar si estos cambios les reportan potencialmente grandes beneficios *a posteriori*, desechando metodologías de tipo paso a paso, tutorizadas o asistidas que pudieran acompañarlos en el desafío.

1 Observar atentamente los comentarios embebidos de Facebook en la siguiente nota: Gallardo, G. (2016). *El calvario de los docentes en Salud Laboral*. Obtenido en <http://www.elsol.com.ar/nota/275223>, [fecha de consulta: 19 de febrero de 2017].

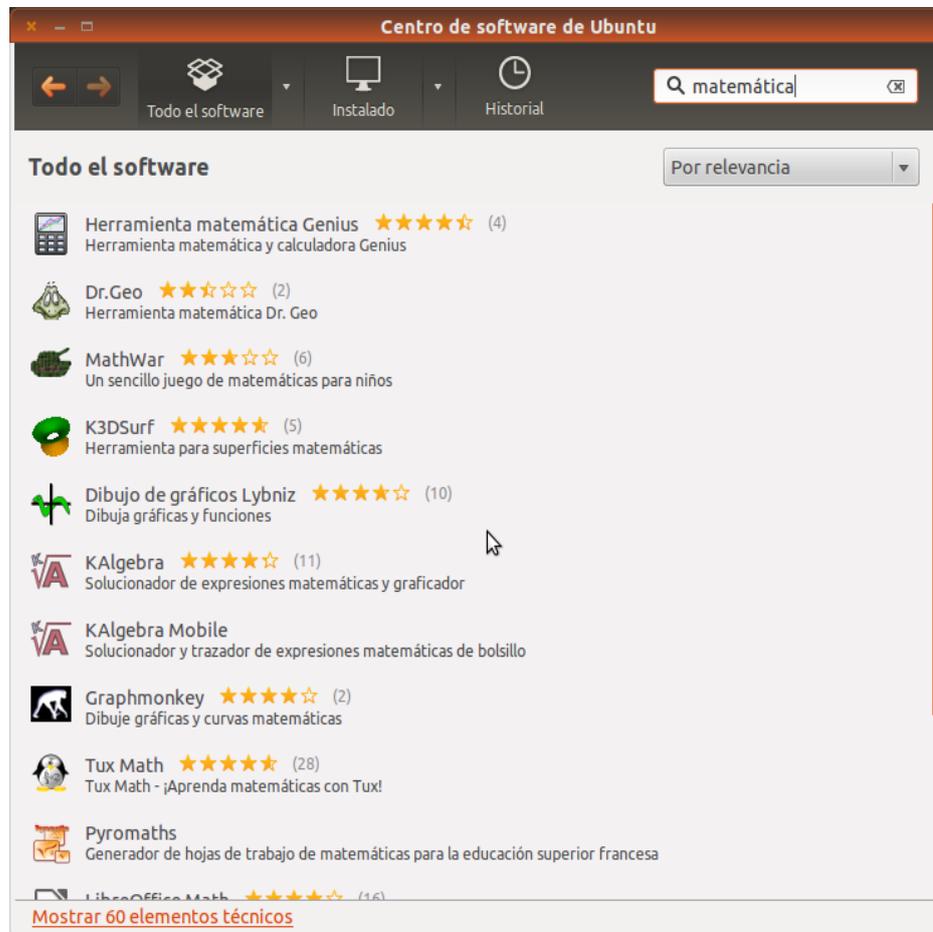


Ilustración A: Centro de software: sitio donde obtener programas, la mayoría gratis

Acompañando esta idea de ser asistidos, la idea de que un *software* puede ser construido socialmente incluso por sus usuarios, provoca no solo uno, sino varios cambios de actitud. Mientras que el docente está acostumbrado a tomar en ocasiones sin permiso *softwares* "terminados" y "cerrados" de grandes compañías, se encuentra de pronto con una cierta anarquía de opciones, incluyendo aquella que consiste simplemente en pedir ayuda. Incluso es válida la pregunta ¿es siempre la libertad la mejor opción?

Un ejemplo de cambio al parecer imposible que los docentes NTIC asimilen es la idea de que el *software* pueda estar organizado en un solo lugar. Es decir, que no tengan necesidad de entrar a la Internet como un "vaquero" y apoderarse como sea de lo que allí encuentre. Es paradójico (y veremos muchas paradojas) el hecho que no necesite actuar de cazador-recolector, ni que tenga que seguir algoritmos de "apropiación", ni que tenga que hacer trampas para obtenerlos. Sin embargo este cambio en la forma de apropiación genera una actitud dubitativa y, muchas veces, va acompañado de la evitación en cuanto al uso.



*Ilustración B: Huayra Linux, Proyecto original de Conectar Igualdad*

Continuando el ejemplo, en Linux, no hace falta meterse en páginas web para bajar los programas, tampoco hace falta "violentarlos"<sup>2</sup>. En un equivalente al Google Play o al Apple Itunes, existe un Centro de *Software* desde donde se puede bajar gratis todo lo que haga falta o, cuanto mucho, a precio de unos pocos

<sup>2</sup> Habitualmente se emplea el anglicismo "cracker", el verbo "crackear" y otras formas derivadas. En este caso hemos decidido traducirlo por "violentar", puesto que "romper" da la idea de que algo, una vez roto, no puede volver a ser utilizado, pero algo a lo cual se le ha violado su seguridad, sí puede seguir siendo usado pese a haber sido violado.

dolares<sup>3</sup> (ver Ilustración A). Curiosamente, a pesar que los docentes entienden el concepto en el sistema operativo Android, ignoran incluso *ex profeso* que la misma idea tiene vigencia desde su creación en 1993 para Debian GNU/Linux, y que se encuentra vigente en la distribución heredada ya preinstalada en las *netbooks*.

Previo estudio exploratorio mediante consulta a informantes claves, encargados de laboratorios y responsables del proyecto Conectar Igualdad, señalan que problemas "bloqueantes" por innovaciones tan simples, no son raras, sino muy comunes. Es decir, existe un esfuerzo muy grande por parte del gobierno en dotar a las computadoras con el sistema operativo Linux y demás *software* libre<sup>4</sup>: esto es, inocuamente instalado en una segunda partición, pero que en la práctica muy pocos aprovechan. La frecuencia con la que se repite el fenómeno da origen a la idea original de la presente investigación, la cual intentará poner en relieve la inquietud de una posible pérdida para el alumno y para el propio docente.

Varios expertos coinciden en que tener acceso a ambos sistemas operativos equivale a contar con una combinación formidable para enseñar. Sin embargo, como en un guiño a la enantiodromía, según la cual todo extremo contiene secretamente su propio contrario, raramente el docente NTIC aprovecha esta ventaja, e incluso sospecha de la misma: dicho con otras palabras: el exceso de tecnología produce un opuesto.

De la misma manera, en el presente escrito efectuaremos un llamado de atención sobre el residuo cognitivo que se pierde, cuando el docente NTIC ignora y desmerece el trabajo voluntario de miles de personas que escriben y ofrecen en

3 Por comparación, la bina Windows – MSOffice cuesta la cantidad de u\$s150 + u\$s750.

4 Se sugiere la visita al sitio del proyecto Huayra:

- Anónimo (2016). *Huayra*. Recuperado de <http://huayra.conectarigualdad.gob.ar/huayra> [fecha de consulta: 9 de marzo de 2016].

forma gratuita sus programas abiertos para aquellos otros que quieran aprovecharlos. Semejante desperdicio no puede tener sino razones más profundas que aquellas obvias que el lector pueda aventurar, razones que se trataran de explorar hasta donde sea posible.

## Objetivos

La presente desagregación de objetivos que penden de los problemas detectados, corresponden a comparar la misión que tienen los docentes y encargados de laboratorios de convertirse en guías y formadores de pensamiento crítico en los educandos, con respecto una realidad no acorde a esta declaración:

- Determinar los tipos de resistencias que manifiestan los docentes de NTIC respecto a los cambios que proponen los softwares libres.
- Examinar factores de resistencia al cambio durante los mecanismos de enseñanza de tecnología e informática.
- Describir los condicionamientos que tienen los docentes de Tecnología y de NTIC, respecto de aquellos docentes de otras áreas que emplean NTIC.
- Analizar la resistencia al software libre como signo de resistencia a novedades por parte de docentes de Tecnología e Informática.

De lo observado y que da origen al problema de investigación, se capta una suerte de idealización sobre la objetividad con la que se enseña, los contenidos que se escogen, y la capacidad de mantenerse actualizado en los sujetos de estudio. Respecto de esta última afirmación, la expresión "actualizado" y la "N" (nuevo) de NTIC van intrínsecamente conectados. La "N" representa un desafío extra que ha sido subestimado, por cuanto nadie puede mantenerse actualizado al minuto, respecto la vertiginosa producción de innovaciones en el campo tecnológico actual, de no mediar acciones permanentes y conscientes, que inciden directamente en el esfuerzo psíquico a emplear en cada resolución de problemas, y en el tiempo que se dedica al fenómeno de la novedad.

Por tal razón es menester que los docentes de NTIC sean al menos sujetos flexibles, adaptables, receptivos y reflexivos al momento de relacionarse con sus propios objetos de conocimiento e intervención educativa. A pesar de que estas cualidades resultan una condición en principio indispensable para este campo curricular, parecen encontrarse ausentes en varios de los casos observados. Paradójicamente, en docentes de otras áreas disciplinares con menor necesidad de relación con aspectos técnicos, la adopción del cambio técnico se da con mayor naturalidad. ¿Cual es la razón? Uno de los principios de la Teoría de Sistemas apunta a que los casos complejos, por su naturaleza, no pueden ser monocausales, sino que por el contrario, son siempre multicausales, y por tal motivo se arriesgan varios intentos de explicación.

## **Hipótesis**

**P<sub>1</sub>: Respecto de los estudiantes y egresados de carreras TIC, ¿es la resistencia al cambio un problema principalmente actitudinal, producto de un engaño primigenio tanto inducido como autosostenido?**

H<sub>1.1</sub>: Las instituciones de formación universitario o no universitaria establecerían claramente las actitudes y aptitudes que se requieren tanto de un estudiante como de un futuro profesional del área o del campo al que se va a dedicar.

H<sub>1.2</sub>: De contar con el suficiente razonamiento lógico-matemático, o al menos una suficiente fuerza de voluntad, los alumnos de las carreras tecnológicas tenderían a finalizar sus estudios, incluso si han perdido toda la pasión que los ha motivado a emprenderla.

H<sub>1.3</sub>: La exposición inicial a interfaces enriquecidas sería directamente proporcional a la resistencia al cambio de frontend, e inversamente proporcional al aprovechamiento del backend subyacente.

**P<sub>2</sub>: ¿Hay una carencia de protocolos previos a la resolución de problemas complejos?**

H<sub>2</sub> Emergente: La Teoría General de Sistemas, así como otros modelos teóricos heurísticos asociados a la complejidad estarían ausentes en la formación de los estudiantes de sistemas

**P<sub>3</sub>: ¿Hay rasgos comunes de resistencia y neofobia en los docentes y los profesionales de informática que dictan NTIC?**

H<sub>3.1</sub>: La resistencia al cambio en los docentes de NTIC se debería a la presencia de estancamiento profesional, rasgos neofóbicos, pensamiento algorítmico lineal que tiende a esclerosarse, escasa investigación, sostenimiento de verdades absolutas, descarte de otras fuentes de información solo por encontrarse estas en otros idiomas y enmascaramiento de su limitado dominio sobre el cambio tecnológico bajo una representación social de aparente actualización.

H<sub>3.2</sub>: El docente de tecnología, cuando se encuentra "atascado", presentaría resistencias al pedido de ayuda, vería afectada su autoestima profesional y tendría resistencia a seguir protocolos éticos cooperativos y colaborativos, si estos lo ponen en evidencia

# Capítulo 1: El perfil como fuente de resistencia

## 1.1. El ingresante autoengañado

¿Cómo llega una persona a tomar la decisión de estudiar una carrera técnica? Quien decide seguir estudios superiores puede verse impulsado por distintos criterios en su elección de carrera. Con aval de lo recabado en las entrevistas, la orientación psicopedagógica de los sujetos responde a diversos factores: 1) imaginarios sociales asociados a representaciones sociales de profesionales exitosos; 2) espacios curriculares de mejor desempeño; 3) pasatiempos (que derivan en un desarrollo profesional); 4) mandatos familiares (el hijo o hija que sigue la carrera del padre porque éste se lo exige o porque tendrá su futuro asegurado siguiendo la línea familiar); 5) impacto de grandes personalidades profesionales y sus modos de vida; 6) campos de alta atraktividad temática; y 7) experiencias laborales con modelos percibidos como de grandes proyecciones futuras.

No es el propósito de la presente tesina indagar en la elección de carreras, en la inserción laboral o en la comparación entre ellas. Pero en general, los estudiantes tienden a admitir que la elección de la carrera tiene que ver, en nuestro caso, con una cierta "familiaridad" con las computadoras. La afinidad de los estudiantes suele estar asociada al uso que le dan en computadoras hogareñas o de la escuela, aplicadas para la recreación con juegos de vídeo o para la redes sociales; los que cuentan con una mayor curiosidad incursionan en la instalación, desinstalación, armado y desarmado de computadoras. Estas experiencias previas los movilizan habitualmente a seguir un proceso de formación sistemática orientándose en un campo profesional y una carrera específica que suponen que satisfará sus inquietudes. Estas inquietudes iniciales se ven estimuladas por el hecho de contar con el acceso a sistemas operativos cuyas

*interfases*, cada vez más amables predicen y facilitan las tareas durante su etapa previa de usuario.

Sin embargo, aquí estamos en presencia de una falacia en la elección de generalización anticipada, que conducen al estudiante novel a la creencia de que las computadoras son fáciles de entender, cuando en realidad son sumamente complejas. No hay razón alguna para suponer que un conjunto inerte de silicio sacudido por ocasionales ráfagas de electrones, controlados por diodos, "compuertas" y unos cuantos algoritmos matemáticos tengan, siquiera por un segundo, la intención de hacernos la vida más fácil.

El hecho de que un programa se torne más "amable", oculta la compleja trama sistémica lógica que lo presenta como simple o manipulable. Y a más amable un programa, más complejo es internamente, porque necesita simular a un ser humano; y uno en verdad muy inteligente. Tales capacidades pueden ejemplificarse en el sentido que la computadora cumple roles parciales de cartero, secretaria ejecutiva, diseñador y matemático, entre otras actividades de asistencia. Y por razones de marketing, especialmente en el mundo de Windows, la simplicidad de la *interfase* no siempre trasunta la complejidad del *hardware* que subyace: por el contrario, todo se presenta como aséptico, edulcorado, sin olvidar la sempiterna palabra "fácil".

En tal sentido ¿puede tanto esfuerzo puesto en la simplicidad, hacer daño? Creemos que si en este caso, respecto del aspirante de las carreras técnicas con elementos de sistemas. Fácil no es lo mismo que simple. Solo por comparación, nunca el profesor de Matemáticas "le miente" a los alumnos dibujando un botón en el pizarrón y luego escribiendo al lado el resultado. El alumno tiene una idea bien clara de lo que le espera en caso que prosiga estudios superiores sobre ciencias exactas. ¿Hay entonces un problema de percepción?

En yoga, Raynaud de la Ferrière al comentar el concepto de *pratyahara* lo define a éste como el "control de las percepciones sensoriales orgánicas, pues el ser humano tiene sus hábitos y un mecanismo nocivo inconsciente" (1971, pp. 53). Es decir, no es nueva la noción que el cerebro, en pos de conservar energía o de otras razones propias del terreno de la neurología, pueda engañarse a sí mismo. Tampoco hay nada nuevo en la realidad de los estudiantes que perciben incorrectamente sus capacidades para afrontar ciertas carreras. Sin embargo, acaso por desconocimiento o por una subestimación en el tratamiento del tema por parte de sus profesores de informática del nivel Medio, el aspirante tenderá a creer que los estudios superiores en informática serán un poco más de lo que hace todos los días (click en Inicio → Programas → Office, etc). Sin embargo, son los usuarios quienes siguen patrones equivalentes todos los días, no el informático: el informático en general es el que crea patrones nuevos para que los usuarios los sigan.

Un ejemplo de programa nocivo que apela al procedimiento automático, a la memoria muscular, al conocido [Siguiete] → [Siguiete] es el virus. Programas capaces de hacer trampa presentándose bajo mensajes inofensivos.

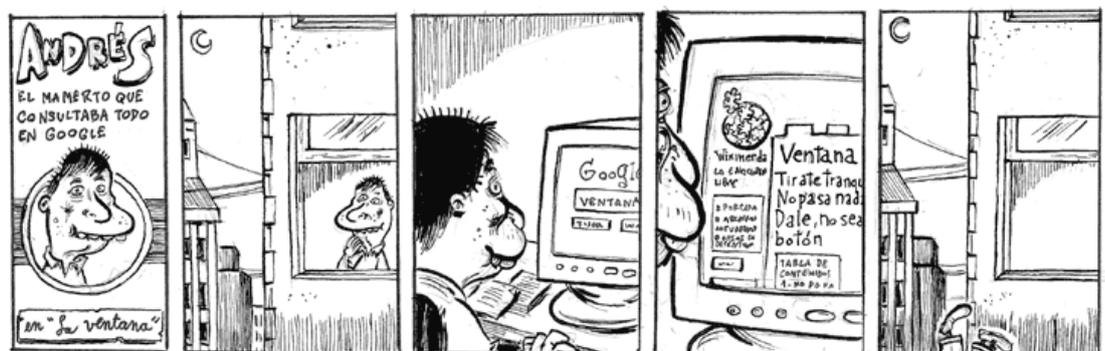


Ilustración C: Podetti en Diario Clarín

El historietista Podetti lo establece muy bien cuando este ciclo aplica a humanos.

Al cabo de uno o dos años de cursado, llega la instancia donde la pasión de los primeros tanteos en el campo de la informática, el análisis de sistemas, la programación y la ingeniería se tornan en un objeto de conocimiento abstruso, difícil de asimilar, complicado para su dominio y obstaculizante para quien quiere hacer de este *métier* su profesión. Lo coherente en esta situación es abandonar la carrera y buscar otra que sea más apropiada.

Este es el punto en el que el estudiante se da cuenta que nunca tendría que haber ingresado. Sin embargo en estas carreras, el autoengaño trabaja a varios niveles; y se autojustifica de formas insospechadas.

### **1.1.1. El examen de ingreso**

Algo que debería haber disuadido al aspirante es el examen de ingreso, ya sea con su modalidad orientadora, nivelatoria, confrontativa o eliminatória, que en teoría advierte a los postulantes respecto de la dificultad que les espera. Sin embargo, muchos estudiantes pasan los exámenes y fracasan después (tras dos o tres años de la carrera).

¿Acaso deberían ser más estrictos los exámenes? No se propone aquí esta idea como válida, puesto que por más difíciles que se presenten las pruebas, éstas no toman en cuenta la voluntad que puede oponer un individuo, en su sueño de llegar a un ideal de profesional bien reeditado, especialmente si los aspirantes se ven inspirados por modelos de éxito monetario en el sentido "¿ser o tener?" planteado por Erich Fromm (2007). Así motivado, un estudiante, tras varios intentos de tipo ensayo y error, terminará aprendiendo todas las respuestas

posibles. Finalmente, pasará todas las pruebas incluso si éstas son extremadamente difíciles.

Por lo tanto, no es un problema de inteligencia, sino de carencia de involucramiento afectivo (pasión) por la carrera. Las *interfases* amables que lo protegieron durante su educación media no estarán disponibles cuando en la universidad, por ejemplo en la materia de Redes, deba configurar una dirección de ip, y mucho menos lo acompañará cuando tras egresar, por ejemplo reemplazar una cabecera de *datagrama*, durante un enmascaramiento entre dos redes.

Aquí es donde el presente documento trata de tomar distancia y evitar hacer una falsa apología respecto del uso de *Linux* en las aulas. Tampoco intenta atacar al sistema operativo de *Microsoft*, sino poner en relevancia el hecho que *Windows* es un sistema operativo que esconde las complejidades, y que se lo aplica en un momento de la vida en que las personas tienen el tiempo para asomarse a otras opciones. Fontanarrosa (1999) lo expresa bien en su aforismo "quiero ser una luz en las tinieblas del conocimiento".

Por tal razón, *Windows* no debería ser usado como único sistema operativo por los estudiantes, porque los trata como usuarios. Y los alumnos *todavía no son usuarios*, en el sentido empresarial del término. Serán usuarios cuando primero formen parte de una elección consciente (no inducida) de lo que les sea útil usar.

### **1.1.2. El perfil equivocado: la habilitación de cada título**

Otro problema recurrente que hace perder el tiempo a los aspirantes a carreras técnicas, es precisamente la elección frente al tráfago de ofertas aparentemente semejantes.

A menudo, las instituciones educativas buscan dilucidar este problema mediante la definición de los perfiles del egresado. Los perfiles deberían advertir e inducir a la reflexión; pero no siempre presentan indicadores indubitables o altamente diferenciados, y no remarcan los límites. Las secciones referidas a los campos de aplicación parecen especificaciones de vehículos 4x4, títulos "todo terreno" que sirven para todos los órdenes de la vida.

Tampoco las parrillas curriculares ayudan demasiado, pues para el lego, determinar que hay más espacios curriculares teóricos que técnicos, no le dice demasiado. Tampoco identificar más espacios curriculares relativos a las ciencias formales y básicas los ayudan a decidirse. Por ejemplo, la mayoría de las universidades coinciden en fijar el perfil de los ingresantes a las ingenierías de la siguiente manera:

"El Ingeniero en Sistemas de Información es un profesional de sólida formación analítica que le permite la interpretación y resolución de problemas, mediante el empleo de metodologías de sistemas y tecnologías de procesamiento de información".

Esta conceptualización es por demás vaga, imprecisa y mal definida, puesto que no resiste el más básico análisis según las siete tradicionales reglas<sup>5</sup> que rige a una definición que se precie de ser científica. Hablar de Ingeniero es hablar de una disciplina tecnológica, no científica, por lo tanto su interés fundamental no radica en la producción de conocimiento científico sino productivo o práctico. En la mayoría de los casos los problemas que resuelven son de esta índole.

5 Las siete reglas son las siguientes: R1 "La definición debe ser más clara que lo definido"; R2 "Lo que se define (*definiens*) debe ser equivalente al concepto a definir (*definiendum*)"; R3 "El *definiens* no debe incluir al *definiendum*"; R4 "La definición no debe expresarse por negaciones"; R5 "La definición debe convenir a todo lo definido y a sólo lo definido (*definiendum*)"; R6 "La definición esencial debe constar de género próximo y de una diferencia específica"; y R7 "La definición debe ser lo más breve posible". Ver Daniel Ruiz (1980), *Filosofía General*. Buenos Aires: Stella, pp. 52 y ss.

Además queda plantear un concepto que ha dado lugar a numerosos debates: el de "información". ¿Qué manejan los ingenieros?, ¿manejan datos o manejan información? El dato es una unidad de percepción; en cambio, la información es una unidad de percepción que adopta un significado para alguien, es decir, un dato procesado por un sistema de codificación susceptible de ser decodificado por otro sistema que cuenta con los procedimientos de decodificación respectivos. Hay quienes incluso cuestionan este ciclo de codificación y decodificación, porque el procesamiento es de tipo primario, pues no hay una resignificación. Los sistemas de significación secundarios realizan este tipo de procesamiento complejo que supone asociaciones no predeterminadas del dato con otro sistema referencial u otros códigos inferenciales (otras lógicas de otros sistemas). Por esta razón se habla del espacio curricular de Sistema de Procesamiento de Datos y no de Información. ¿Por qué se sigue insistiendo en denominaciones de títulos teóricamente inadecuados? ¿Acaso no debería ser el título de "Ingeniero en Sistemas de Datos"?

Asimismo, en las carreras de nivel Superior no universitarias, como la de los Analistas en Sistemas se suele definir (tomando como ejemplo la página *web* del KB - Instituto de Educación Terciaria de General Roca), la siguiente sentencia: "el egresado de la carrera de Analista de Sistemas será capaz de manejar la resolución de problemas complejos en el área de la informática (conocimiento del procesamiento de datos y herramienta de aplicación interdisciplinaria), tanto técnica como metodológicamente, participar en actividades de análisis, diseño e implementación de *software*, así como también interpretar, observar, deducir, analizar y evaluar problemas desde el punto de vista informático".

En ambas carreras se advierte que el egresado vivirá una vida profesional en constante reflexión sobre problemas empíricos, cavilando largamente en torno a problemas técnico-tecnológicos, relativos al procesamiento de datos y sus

tecnologías asociadas. Para ello deberá contar con capacidades especiales de análisis, paciencia, abstracción, introspección y resolución de problemas complejos.

En este caso, la advertencia está puesta de manifiesto. Pero quizás debe ser más explícita. Muchas veces los perfiles son extremadamente simplistas, y se convierten en una herramienta de *marketing*, que busca captar adeptos desorientados, creándoles necesidades o preferencias que no siempre tienen un sustento legítimo. La manipulación cognitiva y volitiva producto de la intervención de los especialistas en mercadeo mina las posibilidades discriminatorias de los buscadores bien intencionados.

Así, para el vulgo e incluso para los estudiantes que están los primeros años, simplemente el ingeniero "estudia mucho". El analista, "menos". El licenciado... "más o menos, agregando contabilidad". La elección de carrera, de acuerdo a lo descubierto en el trabajo de campo, se relaciona más con figuras cercanas de éxito que de comparación entre las propuestas. Ello trae acarreado que los aspirantes, al no leer el plan de estudios o leerlo de manera superficial, supongan toda clase de tareas dispares en su futuro. Así los aspirantes construyen representaciones sociales tales como que "los ingenieros arreglan cosas", "los licenciados programan" y "los analistas sueldan circuitos". Los licenciados, por ejemplo, incluso habiendo egresado, no distinguen la suerte de licencia de investigadores que les habilita la licenciatura (por el trabajo de tesis), ni le ven la utilidad en proseguir estudios doctorales, o en generar conocimiento científico a partir de la gimnasia que aporta el uso de matemáticas o de algoritmos.

Por supuesto, cualquiera de estos profesionales puede ejercer en lo que quiera mientras que esté habilitado. Sin embargo, en la práctica, los analistas disfrutan de la cercanía con el *software*, los técnicos en redes tienden a administrar e interconectar sistemas, los licenciados gestionan proyectos a mayor

escala, los ingenieros en computación y en electromecánica trabajan de cerca el *hardware* y el *firmware*, y los ingenieros en sistemas conectan estos mundos, en forma de proyectos para los cuales deben capacitarse por su cuenta. Hay otras variaciones (Ingeniería de Redes y Telecomunicaciones, Ingeniería en *Software*, Tecnicatura en Programación, etc.) y ninguna es limitante. Pero hay gente que descubre muy tarde el encanto de liderar grupos, poner a prueba programas, "embeber" sistemas operativos, administrar servidores o hacer páginas *web*, y por esta razón deben cambiar de carrera, estudiar contenidos curriculares no requeridos o, peor aún, estudiar de menos.

A falta de una idea clara, los aspirantes recargan las aulas de las ingenierías públicas, sin detenerse a considerar que podrían ser, por ejemplo, felices y satisfechos analistas en sistemas en tres años, o técnicos en redes en dos o profesores en cuatro.

A mitad del segundo decenio del siglo XXI, esta era se conforma con trazos de hiperespecialización y desarrollo tecnológico dentro de cada disciplina. El único consejo que se puede aportar en tal sentido es recordar a los pedagogos durante las clases de orientación vocacional, hacer mención respecto de la diversidad de los perfiles existentes dentro de los mismos perfiles técnicos. Por lo tanto, deberían, o bien documentarse con mayor precisión respecto de la oferta existente, o bien convocar a especialistas de estos campos para que ellos relaten y brinden sus puntos de vista como expertos en el campo.

Permítanos hacer una breve digresión al respecto: los especialistas pueden aportar a los interesados ciertas ideas más concretas de las características del trabajo informático: la ingratitud propia del trabajo informático, por ejemplo, en el sentido que raramente se felicita al trabajador de este rubro por una buena jornada en la que todos los sistemas funcionan sin detenerse.

Otra legalización muy útil sería mencionar el marco relativista en el cual se moverá el egresado: es decir, si bien el trabajador tendrá una jornada laboral de  $n$  horas, en realidad no vivirá en su *tiempo real cronológico*, sino, debido a la concentración y abstracción que le demandan sus tareas, lo hará en un *tiempo relativo einsteniano*. Vivirá buena parte de su vida en un estado que Wolton (2000, pp. 114) lo define como *aplastado*, cuando se refiere a que "este aplastamiento de la duración, esta desaparición de la prueba del tiempo inherente a toda experiencia de comunicación, plantea problemas desde el punto de vista antropológico, puesto que el tiempo de las nuevas tecnologías es *homogéneo, racional y liso*, mientras que el tiempo humano es siempre *discontinuo y diferenciado* [...] un espacio-tiempo que no tiene ninguna relación con la experiencia humana".

El especialista en informática avezado debe transmitir a los aspirantes una toma de conciencia respecto de poder trascender un tiempo lleno de objetos conceptuales que ha de triangular y organizar, así como de reorganizar cuando hay procesamientos anómalos o retardatarios de procesamientos, que toman en consideración sistemas complejos, disipativos y disruptivos, donde las analogías deben ser digitalizadas para poder ser pensadas. Para ello es menester desarrollar un pensamiento sistémico complejo, capaz de pensar complejidades en red que inter-retroactúan, generando cambios que potencian o dificultan la organización, reorganización, distribución y redistribución de datos (*engranajes invisibles dentro de los engranajes*).

La advertencia a los aspirantes debe por fuerza, incluir el hecho que el trabajo en sistemas, muchas veces no está determinado por los horarios pactados previamente. En ocasiones, los trabajos informáticos o de análisis de sistemas requieren tiempos previos de reflexión, de resolución mental de algoritmos y de detección de anomalías técnicas e informáticas, que no se establecen a partir de

un simple ensayo y error. Esos procesos analíticos metaprocesuales exigen un pensamiento sostenido en el tiempo que no suele ser compatible con la organización horaria de las instituciones y sus marcos regulatorios.

La caza de errores lógicos no solo no admite pausas, sino que requiere de una acción sostenida del pensar que no cesa hasta alcanzar la meta. Esta tipicidad en el manejo de los tiempos caracteriza al trabajo del programador, del analista y del ingeniero en sistemas, y es simultáneamente una capacidad estratégica revestida de una apariencia de postergación en función del logro de metas, que de otro modo resultarían inasequibles o susceptibles de ser dilatadas indefinidamente.

Esto se explica de la siguiente manera: un profesional en sistemas no tiene horarios fijos; su organización resulta aleatoria en función de los problemas o tareas emergentes que aborda. A pesar de las buenas intenciones y de las paternales formas actuales que tienen los recursos humanos de tratar a los empleados, para que estos cumplan cuotas "humanas" de trabajo, la realidad indica que el informático *no se va a su casa hasta que termina de reparar el problema*, e incluso *allí será buscado si algo falla fuera de hora*. Toda otra acepción es ingenua. Por un lado, sería ineficiente retomar nuevamente problemas complejos al otro día, tal como el marinero, que no vuelve a puerto hasta que no haya alcanzado el cardumen, so pena de un completo fracaso en su expedición. Por el otro lado, el nivel de urgencia que demandan los usuarios resulta a menudo impostergable, pues los aspectos que antes eran secundarios hoy en día han pasado a ser prioritarios e impostergables (el informático es como un plomero o un electricista que no se va sin dejar reparada la fuga). Cuando el sistema "se cae", se traduce en colas, quejas, angustia, pérdidas de tiempo y dinero para mucha gente.

El aspirante a cualquier carrera de sistemas debe estar advertido que deberá "enchufarse" a la computadora en un nivel de simbiosis analógico–digital que no tiene precedentes en la historia evolutiva humana. Su gran problema radicará en cómo lograr sincronizar sus tiempos personales con los tiempos de la máquina. Aquí hay tiempos personales, tiempos de resolución de problema, tiempos de seguimiento de algoritmos anómalos, tiempos de resolución paralela, tiempos prospectivos idealizados, tiempos de retardo sospechosamente patológicos, etc. La cronología como estudio de los tiempos vividos y percibidos por el informático podrían dar lugar a una interesante tesis doctoral (a saber: la temporalidad en la percepción profesional de los sistemas informáticos e informacionales). Mientras estos tiempos suceden, el profesional debe "sostener una idea en el aire", imaginar procesos, descomponerlos analíticamente, reensamblarlos e inferir anomalías mediante la formulación de universos posibles que han de ser descartados metódicamente, a partir de percepciones precisas y puntuales producto del análisis lógico de los procesamientos y procesos en condiciones regulares o paramétricas. Todo esto requiere tanta fuerza psíquica, como fuerza física necesita un levantador de pesas para mantener un peso en aire durante un cierto tiempo. Requiere asimismo de conocer los propios límites, de la misma forma que un atleta saber cuando conservar energía y cuando consumirla en grandes cantidades.

¿Acaso equivale a decir esto que los informáticos deben ser especies de *mentats*<sup>6</sup> para poder ejercer su profesión? En cierto sentido sí y en cierto otro no. Sabemos que requiere de una buena capacidad de abstracción, tanto intencional como accidental (o inherente a su personalidad). Por ejemplo, para Wolton (2000, pp. 113), "los mejores aprendices son, por una parte, los buenos alumnos y, por el

6 En el universo de Dune —creado por Frank Herbert en su texto de ciencia ficción—, *mentat* designa a un humano adiestrado en el uso computacional de la mente.

otro, el inmenso grupo de personas que tienen dificultades para relacionarse". Muchas personas que sufren de timidez, síndrome de Asperger, variaciones de autismo o sociopatías<sup>7</sup>, son el perfil ideal para esta profesión, y especialmente en el ramo de los QA<sup>8</sup>. Algunas de las personas entrevistadas manifiestan que efectivamente, la informática muchas veces se conforma en un paliativo y en una ayuda a alumnos con patologías como éstas descritas.

Sin embargo, contrariamente lo que la gente cree, los informáticos tienden a ser personas normales, alejadas del cliché de las películas de Hollywood: practican deportes, poseen amistades, cultivan temas espirituales, e incluso incurren en técnicas simples de postura y relajación (*mindfulness, taichi, etc*). Esta heterodoxia de actividades no solo es normal, sino deseable, y debiera asimismo incorporarse su recomendación a la carrera, a la manera que se trata científicamente el tema en profesorados de educación científica. No solo los músculos liberan ácido láctico: también las neuronas se dañan bajo ciertas cargas de trabajo. Desde el momento que se detecta que algunas factorías de *software* sugieren el yoga al personal durante la jornada laboral, surgen interesantes preguntas de investigación relacionadas con posibles problemas en el manejo de la introspección en los informáticos.

7 Peña, I. (2013). *De mendigo a programador en cuatro semanas*, [fecha de consulta, 6 de octubre de 2013]. Disponible en

<http://www.abcdesevilla.es/tecnologia/informatica-software/20131008/abci-programador-sintecho-nueva-york-201310071649.html>

8 QA: Quality Assurance: Personal que controla la calidad de las sucesivas versiones de los productos. Es una rama que requiere tanto de prestar atención a pequeños detalles, como de controlar inocentes cambios que puedan repercutir en otras secciones del producto que se somete a prueba. Llevado a un análisis como el que se realiza en el presente trabajo, las personas con síndromes obsesivos compulsivos tienden a ser bienvenidas en estos empleos, acaso por la percepción gestáltica que contienen sus observaciones: tema interesante a desarrollar en otras investigaciones.

### 1.1.3. La habilitación, ¿genera competencias para...?

La universidad aporta al menos tres contenidos que utilizará el profesional en su quehacer: los contenidos conceptuales (conceptos, definiciones, teorías, modelos teóricos, teoremas, explicaciones, cuerpos de hipótesis en boga, etc.), contenidos procedimentales (técnicas, estrategias, métodos, actividades típicas, etc.) y contenidos actitudinales (habilidades, destrezas, capacidades, actitudes, valores, principios de actuación, normas, códigos de actuación, etc.) tales que permiten adquirir nuevo conocimiento. Dependiendo de las carreras, es mayor o menor el aporte de estos contenidos. En el caso de sistemas, lo procedimental tiene un componente significativo, impulsado incluso por los mismos estudiantes, quienes se consideran individuos más prácticos que teóricos. Sin embargo, deben lidiar con mucha teoría antes de poder pasar a la práctica. Luego volveremos a este tema, pero a este nivel baste considerar que "todo" estudiante de sistemas aprende de la manera dolorosa, cuando programa atropelladamente sus primeros algoritmos contra bases de datos, sin detenerse a reflexionar los elementos básicos (teóricos) de, por ejemplo, *normalización de datos* y creación correcta de *diagramas DER* (Diagramas de Entidad-Relación). El castigo es tener que reprogramar nuevamente la mayor parte de la aplicación.

Aquí es donde la actitud tiene más valor que las competencias en los otros dos aspectos referidos (procedimentales y conceptuales). La actitud de estar pronto y receptivo a toda nueva información debe ser *previa a todo*. El alumno debe estar preparado para simular en su mente que el tema "es apasionante" *antes* incluso de abordar su ejecución. Y esta gimnasia de encarar los desafíos no se termina cuando la casa de estudios lo haya declarado habilitado (cuando reciba el título) sino que será lo más parecido a una herramienta (aquí la actitud pasa a ser la herramienta, no el *software*); aquella que lo distinguirá como un profesional

competente, aquel individuo capaz de abordar temas, que en su mayoría, serán nuevos incluso si han pasado unos pocos años desde la última vez que los pudo dominar. Es decir, reaprender temas que todo el tiempo serán *nuevamente nuevos*.

Aquí es donde radica la importancia de tomar consciencia que el perfil del informático requiere de personas mentalmente inquietas por lo nuevo, sin importar si es de índole teórica o práctica.

Por comparación, cabe pensar que en caso de seguir un Profesorado de Historia, las distintas eras geológicas seguirán vigentes y que, en un futuro cercano, no deberíamos esperar grandes cambios al respecto. Si el egresado de una carrera no posee grandes inquietudes intelectuales en el campo de la investigación, "puede vivir" de esos conocimientos troncales, aplicándolos en diversos quehaceres de su actividad profesional y transmitirlos a un público lego tal cual como le fueron transferidos. Inclusive puede tratarse de un catedrático brillante, que aclara conceptos de alta complejidad, pero que solo han sido aprendidos en su formación de grado<sup>9</sup>.

El informático también puede vivir sin inquietudes por lo nuevo. Pero vivirá mal. Técnicamente, apenas sobrevivirá. Y es así por que en el campo de las NTIC es un imperativo la actualización constante en temas nuevos y desconocidos, incluso "antes" de transferirlos a un público lego en la materia; he aquí precisamente la gran diferencia con otros espacios de conocimiento, sobre todo si nuestro individuo de ejemplo se decanta por la docencia: sus educandos

9 Ver nota: "Derecho renueva su plan de estudios por primera vez en 25 años en la UNCuyo." Anónimo (2016). *Derecho renueva su plan de estudios por primera vez en 25 años en la UNCuyo*. Recuperado de <http://www.diariouno.com.ar/mendoza/derecho-renueva-su-plan-estudios-primera-vez-25-anos-la-uncuyo-20160521-n793212>, [fecha de consulta el 22 de mayo de 2016].

rápidamente percibirán la obsolescencia de sus enseñanzas. A diferencia de las "eras geológicas", varios de los conceptos que se enseñan en TIC, simplemente no existieron hasta hace unos pocos años. Y además, todo el tiempo se van *refactorizando* (en el argot informático; reformulando, rediseñando e innovando). A la fecha, tal es el caso de tecnologías como monedas virtuales, Docker, NOSQL, o microservicios, que apenas hace unos años eran un esbozo y hoy son una realidad.

En las carreras técnicas la base que aporta la universidad cubre apenas lo necesario para construir las bases del accionar profesional. Así debería ser entendido y advertido desde el momento en que aspirante inicia su universitario. Es sabido que la información técnica de la que se dispone queda obsoleta al cabo de tres años, que es el tiempo que tarda en salir la versión nueva de un software. Este tiempo, actualmente en aceleración debido a la adopción de modelos "*rolling release*", una modalidad en la cual se deja de mantener versiones viejas de software: conceptos provenientes de Gentoo, FreeBSD y de Archlinux, y actualmente instalándose en el software cerrado, como Microsoft, cuyo producto Windows 10, una vez arribado a una cierta estabilidad ha declarado obsoletas todas las versiones anteriores del sistema operativo. En términos generales, todo lo escrito en informática queda desactualizado prácticamente al llegar a la planta impresora, e incluso el argot empleado en la industria evoluciona cada año<sup>10</sup>.

Continuando lo expresado, en el caso de Argentina hace solo seis años no se hablaba de profesiones como *DevOp*, *Quality Assurance*, Domótica, Mecatrónica, *Community Manager*, o Analistas SEO (*Search Engine Optimizer*).

10 Curiosamente, la formación de base que aportan los sistemas POSIX o SAP poseen una trascendencia mayor en el tiempo. Cambian las *interfases*, pero la arquitectura subyacente de *mainframes* que sostienen Internet y su red de servicios es prácticamente la misma que hace 30 años. Las viejas revistas sobre Unix y Linux, a la manera de los viejos griegos, son completamente vigentes.

Incluso el intento de traducir al español estos puestos de trabajo tiene poco sentido y menor adopción aún, por el rápido reciclaje que se hace de los mismos en el mercado actual. Es decir: para el momento en que la traducción adquiere el mismo sentido (o "*meaning*"), el término ha quedado obsoleto en su propia lengua.

Solo la disciplina y la costumbre de investigar es la actitud fundamental a desarrollar en el egresado. El informático debería ser formado como un investigador-actualizador, para que acostumbre a no dar nada por supuesto.

Incluso la acreditación en informática solo inspira una tibia confianza en las empresas respecto de que el egresado podrá afrontar ciertos cambios, pero en modo alguno asegura las competencias profesionales del mismo. El egresado debe trabajar duramente para cubrir la brecha, porque las leyes del mercado así lo exigen. De esta manera, no tiene sentido que la universidad lo entrene en los 20 lenguajes de moda<sup>11</sup>, sino que le aporte una buena base de algoritmos.

Cuando el cambio arrincona a los profesionales de sistemas, estos tienden a atribuir sus fracasos en el presente a deficiencias en el pasado durante la formación académica. Si bien no se trabajó esta línea, varios testimonios semejantes que coinciden con esta aseveración pueden encontrarse en las entrevistas que se encuentran en el Anexo 1 del CD que acompaña el presente trabajo. Sin embargo, de lo observado se detecta que los contenidos se hacen obsoletos incluso mientras se los está cursando: lo contemplado en primer año como cualquier proyecto "de innovación" serán "realidades instaladas" ya en su quinto año. Cuando egrese, no serán desafíos futuros, sino etapas teóricamente asimiladas de su pasado. Esto queda demostrado cuando el alumno una vez egresado, poco le dura el alivio de dejar los libros: la realidad de la profesión lo

11 Gráficas en tiempo real sobre los lenguajes más utilizados:

Anónimo (2016). *TiIOBE Index*. Recuperado de <http://www.tiobe.com/tiobe-index>, [fecha de consulta: 9 de febrero de 2017].

vuelve a situar en la incómoda situación respecto que *siempre* deberá adelantarse en el futuro respecto de sus contemporáneos.

Este estado de "yecto" heideggeriano (Heidegger, [1927], cap. V, § 38) indica el carácter mismo de aperturidad del sujeto contemporáneo y, por antonomasia, aquel ser humano que está sujeto a la esencia de la técnica, aunque sin la carga negativa que le otorga el autor. Él sujeto, en nuestro caso, objeto de estudio, vive una realidad ampliada. Es decir, su presente se extiende a las proyecciones posibles, deseables y factibles de los escenarios futuros que se puedan prospectar. Vivir el presente en este vertiginoso campo es una forma de vivir el futuro: una pseudorealidad bastante diferente a la que imaginaron sus padres.

En un futuro utópico las computadoras "nos traerían el desayuno a la cama". Y no es que no puedan hacerlo: pero debemos programarlas para ello previamente. No obstante, incluso siendo muy "tontas", nos transportan al trabajo, nos sugieren que cosas comprar e, inclusive, que amigos frecuentar<sup>12</sup>.

Si la universidad hiciera todos los esfuerzos para que sus estudiantes adquirieran las reales competencias que hacen falta para cambiar el presente actual, continuo y estatificado hacia la utopía que fue alguna vez, probablemente no tendría egresados, ya que los planes de estudio agregarían cientos de espacios curriculares complementarios.

Intuitivamente los jóvenes saben esto: son de hecho muy buenos ignorando las experiencias de los expertos, mientras seleccionan al azar aquello que es inmediatamente útil. La naturaleza es sabia y exigente, e induce a una rápida

12 Son ejemplos cotidianos el uso cotidiano del GPS de las computadoras de los autos y de los teléfonos celulares. También de las predicciones de compras vía *cookies*, o de los algoritmos sociales de Facebook basados en distancia geodésica, por mencionar solo algunas de las formas en que el *software* nos cambia la vida.

competencia en capacidad y velocidad de selección de los objetos disponibles. En lugar de indignarnos por el desaire que hacen los jóvenes de nuestras extensas reflexiones y peroratas, deberíamos limitarnos a señalarles cuándo algo gratis es "bueno", respecto de cuándo es gratis solo en apariencia, pero que distrae como un espejismo al sediento que vagabundea por el desierto. En el caso del informático, es necesario distinguir al cazador-recolector respecto del carroñero cleptoparásito, comportamiento que acaso compartimos con el *homo heidelbergensis*. Para el caso de estudio que nos compete, el docente de informática / encargado de laboratorio, cuando consume software robado cada vez que puede, para ahorrarse en términos de energía psíquica la programación o al menos el *fork* de proyectos libres. A fin de no perder el foco ni la objetividad, esta línea de investigación se desarrolla aparte, en el apéndice E.1.2, con bases suficientes como para ser continuadas por aquel lector que le parezca aplicable en otro trabajo más relacionado con antropología, psicología o sociología.

De lo anterior baste mencionar que los docentes deberían señalar a los estudiantes la diferencia entre "gratis" y "abierto" como un deber profesional entre sus competencias, haciendo honor a la responsabilidad que significa opinar correctamente sobre tecnología de base. Ello supone una formación según criterios, lo cual brilla por su ausencia en los contextos educativos contemporáneos, a pesar del sinnúmero de proclamas pedagogizantes que reinan en el medio, plagada de reduccionismos mal hechos. Para Watzlawick (1992, pp. 61) las simplificaciones son peligrosas, por cuanto la complejidad es en gran medida irreductible. Simplificarla es una tarea fútil e ingenua, por cuanto arrastra a bucles que niegan la realidad (1992, pp. 64). Persistir en la negación de la complejidad conduce a la negación de la propia negación en sí misma, un efecto Coriolis que solo puede llevar a la intensificación del problema: el desconocimiento *ex professo* de no distinguir programas gratis de abiertos, es ofrecer voluntariamente la vida personal al *pentesting* de los *black hat*, ingenieros sociales

y conocedores por excelencia de la indolencia humana. Tal es la razón por la cual la Teoría General de Sistemas —la teoría que no se estudia adecuadamente en sistemas, ya volveremos a este tema— prefiere enfocarse en estrategias para lidiar con la complejidad, en lugar de subestimarla.

## **1.2. El alumno desapasionado a mitad de carrera**

Se había subrayado anteriormente la necesidad que tienen las profesiones técnicas de actitudes tales como pasión y entereza. Es frecuente el hecho de observar que estudiantes de carreras técnicas, hacia la mitad del cursado, entran en un estado de desapasionamiento completo; no incluyo aquí a aquellos a quienes los contenidos les resultan muy complejos, sino a aquellos que simplemente no se imaginan trabajando en lo que estudian, o que son conscientes de que se aburrirán muchísimo en el desempeño del mismo. Se trata de una visión fatídica de un futuro que muestra escenarios y modos de actuación no deseados o con los cuales no encuentra una satisfacción o sentido de plenitud. El sentido de autorrealización del sujeto se encuentra en crisis.

Como decía Nietzsche en su *Origen de la tragedia griega*, cuando uno se encuentra leyendo a la mitad la *Ilíada* o la *Odisea*, se vivencia un sentido de infinitud angustiante: por delante parece no haber una orilla, por detrás menos aún. Es así que el sujeto tiene que tomar una decisión crucial: sigue adelante, termina y asume su futura profesión, o renuncia y deja todo lo realizado como un ciclo no resuelto, como una deuda pendiente de insatisfacción, una frustración ante lo no logrado y una inversión económica que nunca dará sus dividendos.

En efecto, un estudiante de medicina que se maree en la morgue y que se aburra en clase, será consciente de que no estará a la altura de asegurar la vida

de sus futuros pacientes. Sin embargo, ¿cuánto daño puede hacer un informático, un analista, un ingeniero desapasionado si se atreve a terminar? Esta discusión es fascinante ya que se puede transpolar a múltiples niveles. La letalidad de un programa mal hecho, normalmente se limita a los datos del usuario. Datos que generalmente pueden volver a recrearse. Además, un informático desapasionado probablemente no posea un *currículum* tal que sea elegido para ajustar algún día el ángulo de reingreso de una cápsula tripulada desde el espacio. Solo en raras ocasiones, y sin pretensiones de caer en la ironía, la falta de atención y de compromiso provoca errores grandes, como el despegue del Challenger, el aterrizaje de la sonda Schiaparelli, o los mecanismos de enfriamiento de la planta Fukushima.

El alumno desapasionado, pero aún competente en lógica y matemática, se propone hacer un mayor esfuerzo y terminar "para no seguir estudiando nunca más", incluso a sabiendas de que estas profesiones se encuentran entre las que requieren de mayor actualización permanente. Lo interesante del caso es que el alumno ya se había autoengañado antes, durante el examen de ingreso, quizás inconscientemente. Esta segunda etapa de autoengaño es por el contrario, consciente, con un propósito nuevo: vivir de algo que no le gusta, con toda la actitud de servicio que pueda adjuntar a este (des)propósito de vida.

Eventualmente el alumno obtendrá su título habilitante. Ante la ausencia de un organismo colegiado que controle —y para desgracia de sus futuros clientes—, su desapasionamiento quedará inadvertido, los datos expuestos y los procesos desbocados, siguiendo una evolución caótica o al menos aleatoria sobre las superficies de los infortunados discos duros que se crucen a su paso.

### 1.2.1. El egresado lisiado

Habíamos mencionado que muchos ingresantes escogen estas carreras alentados por la facilidad de uso de las *interfases* gráficas, ventanas plenas de elementos "enriquecidos", esto es, cuadros de control fáciles de operar e intuitivos. A modo de ejemplo, una caja de texto en una página Web que no permite el envío de un formulario, inhibiendo la creación de una cuenta de usuario, recordándole al usuario que su correo electrónico ya existe en la base de datos, y sin realizar una recarga completa de la página completa, parece natural y su comportamiento obvio; sin embargo es extremadamente compleja por dentro, puesto que requiere de varios conocimientos de Javascript, CSS y HTML, sumado a algún lenguaje del lado del servidor de aplicaciones, y de otro lenguaje para recabar los datos desde la base. Cinco lenguajes solo para que un botón funcione.

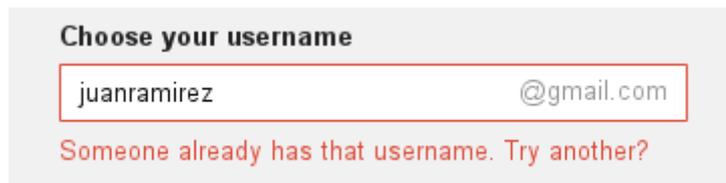


Figura D: Creación de cuenta de usuario en Google

Aquí es donde el egresado pone en evidencia tanto sus logros (destrezas, habilidades, pensamiento algorítmico y sistémico, competencias y talentos), como sus falencias (impaciencias, capacidades heurísticas) cuando intenta resolver con *interfases* gráficas enriquecidas y amables algo que debe ser programado usando a la vez lenguajes de programación y línea de comandos. Todo para en definitiva, crear una simulación *cyborg* de un comportamiento humano. Comportamiento que por su naturaleza es extremadamente difícil de simular y de predecir.

Aquellas personas que se enojan frente a un componente enriquecido como el ejemplificado, ciertamente no toman en cuenta las miles de horas que llevan su

realización, desde la conformación de las librerías hasta la adecuación del protocolo, su validación y prueba. El usuario final normalmente no reflexiona respecto de las miles de personas interesadas y trabajando por detrás en que *él pueda loguearse fácilmente*<sup>13</sup>.

Aquí es donde el egresado choca. ¿Cómo puede ser que una pequeña caja de *login* sea tan difícil de programar? El egresado se encuentra inerme cuando, a pesar de su título, no puede acometer los problemas que le encargan resolver. A menudo, los ingresantes a carreras informáticas del área caen en una suerte de autoengaño al creer que, por el solo hecho de a) coincidir con un patrón de comportamiento de los programadores que hacen *interfases* gráficas y b) hacer intuitivamente lo que se esperaba que hiciera —usar una computadora sin dificultades—, poseen habilidades y capacidades por encima de la media. Este pensamiento simplificador y pueril adolece de una comprensión más profunda del problema, a saber: existen múltiples modos de pensar y de relacionarse con las cosas. Los programadores evidencian un tipo de pensamiento que no siempre es seguido por el resto de las personas, pero eso no significa que su pensamiento sea superior o que por su complejidad se encuentre limitado a unos pocos privilegiados, sino simplemente que existen otros modos de pensar, según otros principios organizacionales que pueden llevar a incompatibilidades interparadigmáticas o interlógicas, que hacen que para algunas personas, la obviedad del programador sea lo menos obvio.

En medio del cursado, el estudiante debió darse cuenta que la informática es un tema bastante complejo de fondo. A pesar de ello siguió cursando con la

13 Es paradójico que el usuario, impaciente ante una respuesta inesperada, sospeche que "del otro lado" no lo deseen adentro, o que la computadora ha sido programada con una animosidad especial contra él. En la figura D —trasladable a cualquier otro servicio de correo o red social existente—, miles de programadores dedican sus vidas a que esos formularios no fallen, por la sencilla razón de que pierden clientes.

esperanza de algún día superar este tipo de obstáculos. Puesto que todos somos rescatados por *Google* casi todos los días, se podría decir que este buscador es el principal propagador de la "fe" informática.

Sin embargo no todo está en *Google*. Ciertas respuestas requieren de descubrir la respuesta sin ayuda. Aquí es donde se detecta la falta de uso de *software* libre, o al menos de CLI (*Command Line interfase*) antes de ingresar a la universidad. Si al menos una vez el estudiante antes de ingresar a la universidad hubiera tenido que abrir la "tapa del motor", se hubiera dado cuenta de la seriedad del asunto. Si al menos en algunas ocasiones hubiera usado el Símbolo de Sistema (MSDOS) de *Windows*, se habría aproximado al oscuro bajo fondo que subyace a sus "ventanas". Esta es la razón por la cual se incluye programación en la currícula de los estudiantes de secundaria en el Reino Unido<sup>14</sup>. En estas planificaciones se trasciende la visión de sus creadores, quienes apuntan a que el simple hecho de compilar algún pequeño programa tal que instrumente al menos contadores y sumadores, supone una visión previa mucho más lucida, clara y objetiva respecto de lo que se puede esperar en caso de plantearse estudios de informática.

### **1.2.2. El egresado competente... en otros campos**

Muchos especialistas en supervivencia concuerdan en la necesidad de al menos dos de los siguientes tres factores fundamentales para el éxito: habilidad, experiencia y suerte. La recontextualización y traspolación de esta tríada se aplica perfectamente en la ciudad ("la jungla de cemento") y específicamente en el departamento de IT ("Information Technology").

<sup>14</sup> Seguir la siguiente nota escrita por Livingstone, I (2013). *Teach children how to write computer programs*. Recuperado de <http://www.theguardian.com/commentisfree/2012/jan/11/teach-children-computer-programmes>, [fecha de consulta: 6 de Octubre de 2013].

Sabemos que el egresado no puede alegar gran experiencia, de modo que solo la universidad puede asegurar que tiene la capacidad para realizar múltiples cosas, procesos y acciones asociadas a su competencia profesional general y específica. En teoría, si el egresado pudo con todas las materias del plan de estudios, probablemente pueda resolver otros problemas también. Personas así garantizadas por su universidad, son suficientemente aptas para muchas empresas.

Sumando el hecho que estos egresados todavía no han asumido un conjunto de tareas profesionales que les "formatee" sus capacidades, muchas empresas captan estudiantes de ingeniería antes de recibirse. En un mundo tan cambiante, donde la experiencia adquirida en el primer año probablemente haya quedado obsoleta en el quinto, entonces al menos podrán contar con un perfil capaz de improvisar sobre la marcha. Esto sería una habilidad "virtual" que le adjuntan, ya que el estudiante no puede demostrar todavía su efectividad en el cargo.

Esta afirmación respecto de la importancia de la habilidad antes que cualquiera de los otros factores, se encuentra en la persistencia de este modelo de incorporación de recursos humanos. Los esfuerzos que se toman los encargados de recursos humanos en seguir y convencer a los egresados de las ingenierías, produce una buena tasa de retorno con el tiempo. En el peor de los casos las empresas consiguen, si no buenos ingenieros, al menos buenos empleados para otra cosa. Los ingenieros son excelentes en áreas como Compras, Logística y Proyectos. Probablemente no pasen de usar una hoja de cálculo, el *Microsoft Project* y el cliente de correo, pero serán muy competentes y probablemente saquen adelante cualquier departamento que encabecen. Así lo esperan varios estudiantes de ingeniería, quienes vienen aprendiendo sobre gestión de proyectos ya sobre la misma carrera.

Entonces, si se descarta la experiencia como factor de contratación de egresados, y se deja sólo a la habilidad y a la constancia como factores preferidos, es simple entender la razón por la cual las mismas empresas no buscan egresados de Análisis en Sistemas o Licenciados, sino Ingenieros. No es tan difícil sobrevivir en carreras de grado 3 o 4 años, en comparación con las Ingenierías, más nutridas de materias técnicas, lógicas y matemáticas. Por esta razón los egresados de Análisis en Sistemas o Licenciaturas deben recorrer un camino mayor (postítulos), a fin de adquirir conocimientos y exhibir una foja de servicios tal que les permita aspirar a los mismos cargos: doctorados, maestrías o, al menos, una fuerte experiencia.

### **1.3. El mito de la elección consciente del software, y sus consecuencias en los docentes de informática y tecnología**

Volviendo al estado de "yecto" heideggeriano (Heidegger, [1927], cap. V, § 38), el egresado devenido en docente de tecnología, en un estado de aparente razonamiento circular, tiende a defender y a recomendar las *interfases* de *software* comercial, aduciendo la aceptación popular mayoritaria como único argumento de autolegitimación. Si se lo confronta a justificar racionalmente su argumento mediante juicios completos de valor respecto del *backend* subyacente al *frontend*, se abstiene o se retira. Este comportamiento es alarmante desde el momento en que la defensa la realiza desde la posición de un cliente fidelizado, convencido de recomendar un producto basándose en la creencia de una libre elección previa. Ello es propio de los ciborgos o *cyborgs*, seres humanos con identidades construidas tecnológicamente, que viven en un mundo simulado. Según Burbules (2001, pp. 210),

"[...] el uso creciente de grupos focales y de técnicas de encuestas, así como el diseño sofisticado de ordenadores, permitieron a los publicistas, agentes de marketing, encuestadores y políticos ser cada vez más expertos en desarrollar perfiles del público o de los consumidores, a fin de anticipar las preferencias de las personas, antes de que éstas hayan considerado siquiera las elecciones mismas que pudieran efectuar".

Más adelante destaca que

[...] el producto es una simulación destinada a satisfacer las expectativas del público, el cual está constituido por consumidores que ya se sometieron a encuestas en otras oportunidades, y cuyas preferencias y reacciones se conocen de antemano: los productos se diseñaron de acuerdo con ellas. Uno cree que como consumidor (o votante, etc) hace una elección; empero, esa elección está preparada para que uno la haga así (y por supuesto que a uno le gusta, ¿cómo podría ser de otro modo?).

Volviendo a la defensa que hacen los docentes de tecnología respecto de usar siempre las mismas cosas (solo *Windows* y *Microsoft Office*), se trae a colación un artículo de Heinz von Foerster, intitulado "Notas para una epistemología de los objetos vivientes", sobre el cual Marcelo Pakman comenta que "toda teoría que da por sentado lo que pretende explicar, termina siendo una tecnología, o bien, podríamos también agregar, una ideología" (2006, pp. 38). Si esta defensa se realiza honestamente partiendo de una visión de homogeneidad en las herramientas que usan los educandos, es cuando menos ingenua pero válida. De otra manera es simplemente un dogma.

### **1.3.1. El software como una nueva zona de confort**

La discusión en este punto se traslada a la importancia que parece tener para los docentes de tecnología, y para los usuarios por extensión, respecto de la importancia de mantenerse a toda costa en la zona de confort. Y me apresuro a señalar que la zona de confort, en materia de educación, no es la mejor zona

donde estar: es solo una base. Los alumnos saben muy bien del tema porque todo el tiempo los docentes los han de sacar de su zona de confort. De otra manera es cuestionable que aprendan por *motu proprio*. ¿Y con los docentes de tecnología? ¿Cómo se hace para que los docentes sigan aprendiendo? Difícil es convencer a un individuo que aprenda cosas nuevas cuando éste considera que ya sabe "todo lo necesario".

Las verdaderas razones parecen encontrarse en las mismas raíces de la informática; en las palabras de un médico y neurólogo inglés, que contribuyó decisivamente a la consolidación de la cibernética moderna, llamado Ron Ashby, inventor del homeostato. En sus observaciones sobre las máquinas ultraestables, Ashby (1965) refiere a que el medio y las herramientas condicionan la reacción del individuo. De no haber un cambio en el medio, el individuo no cambia. O al menos no cambia, mientras su versión mental del homeostato se mantenga dentro de valores normales.

El docente de tecnología, quien mantiene en un estado semejante de homeostasis a sus construcciones mentales, habiendo llegado a un cierto nivel de estabilidad, enfatiza el no cambio (estaticidad inercial). Por ejemplo, enaltecen el costo altísimo de uso de CPU que hacen a *Windows* más estable, gracias a que éste "atrapa" sistemáticamente toda excepción posible. Luego de años de "cuelgues" y errores, se ha arribado a una modesta estabilidad; ya no hace falta refactorizar la deficiente y heredada arquitectura original, basada en el extraño concepto de no permitir el borrado de un archivo mientras éste se encuentre en ejecución, cuando podrían ser estos archivos reapuntados como tuberías (*pipes*) desde la memoria —concepto tan viejo como Unix (1969).

Este aspecto es bien conocido por los usuarios de *Linux*, "pingüinos" que raramente necesitan reiniciar o "dar vuelta" el iceberg luego de haber descargado actualizaciones.

El paradigma central ha sido salvado mediante una constelación de hipótesis auxiliares, que operan como un sistema de muletas capaces de recuperar la capacidad de locomoción damnificada luego de un accidente. El sueño del homeostato de Ashby es defendido de todo rayo gamma que pueda romper las cadenas helicoidales de una arquitectura apresurada, creada para monousuarios y apuntalada para dar cabida a TCP/IP y un entorno multiusuario, tal que resulte sostenido gracias a velocidades de los procesadores actuales y por profesores que, en su confusión, defienden la novedad de terminar su trabajo diario sin cuelgues: el *software* hace lo que debió hacer hace veinte años. Se trata de un *software* que sirve "hasta para trabajar", *sin miedo*. Por ejemplo, el viejo tic de pulsar compulsivamente Ctrl+G (atajo para guardar) ha ido desapareciendo paulatinamente hasta convertirse en un oscuro miedo atávico, cuyas bases retrotraen a la noche de los '90.

Toda nueva innovación al respecto acarrea la preocupación de cambiar aquello que tardamos tanto en conseguir. Ashby (1965, pp. 57) se refiere a esta complejidad señalando que "*conforme aumenta el número de variables, normalmente es preciso coordinar los efectos que cada una ejerza sobre las demás, con un cuidado creciente, para que pueda conseguirse (nuevamente) la estabilidad*"<sup>15</sup>.

La estabilidad basada en la retrocompatibilidad no es un tema menor. Si la gente decidiera usar *Linux* de la noche a la mañana, descubriría que no todas las aplicaciones —al menos, no con seguridad la de *Microsoft*— funcionan, lo mismo ocurre con algunas tales como *Adobe* y *Autodesk*, muy arraigadas en la masa

15 En inglés, el original: "*here it is sufficient to note that as the number of variables increases so usually do the effects of variable on variable have to be coordinated with more and more care if stability is to be achieved.*"

consumidora. Dichas aplicaciones tienen mucho peso al momento de optar por una plataforma.

En este punto es importante señalar a aquellos docentes que no se encuentran relacionados con tecnología, personas que no poseen prejuicios. De hecho, cuanto menos saben, más fácilmente aceptan los cambios, *porque no saben que se trata de un cambio*. A mayor ignorancia, menor resistencia. Sólo cuando les es señalado que son diferentes, se pliegan a la opinión popular y solicitan les sea desinstalado *Linux*, a veces sin siquiera probarlo.

Muchas veces la opinión procede de otros docentes que dicen no poder usar adecuadamente *Linux* u *Openoffice*. La experiencia indica que estos docentes tampoco usan adecuadamente *Windows*; basta con observar sus sistemas contaminados e infrausados. Un docente que asegura "necesitar" *Microsoft Office* usualmente no alcanza a usar el 20% de las opciones del menú, e incluso ni siquiera usa el menú, pues sólo se limita a los iconos visibles. El menú es un área oscura y aparentemente infinita; es la razón por la cual la interfaz *Fluent* (de *Office 2007* y versiones posteriores) esté compuesta por puros iconos; un cambio no del todo descabellado, pero que provoca más consternación en los viejos docentes, que productividad.

Lo que sí se puede demostrar, en cambio, es la ventaja competitiva para sus diseñadores, quienes introducen una incompatibilidad transversal para los nuevos usuarios respecto de otras propuestas. Estos usuarios —jóvenes— son los que importan en términos económicos a futuro. Idealmente cautivos, no podrán "cruzarse de vereda", y la competencia, detenida por el *copyright* de la nueva *interfase*, tampoco podrá ofrecer una alternativa semejante hasta encontrar un hueco legal. Esto toma años y dinero a las cooperativas, fundaciones y universidades que amparan el uso del *software* libre y que están lejos de poder destinar fondos al respecto.

De esta manera podemos enunciar que la plasticidad mental es directamente proporcional al uso continuado de una misma herramienta. Las herramientas vistas como un objeto, al dejarse trascender, contaminan al sujeto que las estudia. Por esta razón los docentes de otras tecnologías, con una falta de referencias previas, pueden imaginar el problema desde varios ángulos, o al menos escoger sin prejuicios otras herramientas.

El otro aspecto es la ausencia de condiciones epistemológicas con las que el docente acompaña su recomendación sobre usar *software* cerrado. El único razonamiento ofrecido, consiste en que hay que enseñar lo que usa la mayoría. Este argumento, sumado a que en teoría, se encuentra más capacitado para señalar el camino correcto, hace que su opinión tenga más peso que la de un usuario, cuando en realidad lo único que hace es gala de una vieja falacia conocida como *argumentum ad populum*, y desde su figura de conocedor aprovecharse para manipular a los legos en función de ahorrarse el aprendizaje de conceptos nuevos.

En todo caso, su recomendación sería objetiva y válida en tanto fuera capaz de usar con soltura cualquier tipo de *software*: abierto o cerrado. Y la evidencia encontrada durante la migración a Linux en la Escuela Normal 9-002 "Tomás Godoy Cruz", indica que los 7 sujetos estudiados se negaron a asistir a las capacitaciones realizadas con motivo de la migración del laboratorio, fueron incapaces de usar *software* libre incluso para las tareas más elementales, y finalmente se resistieron en todas las ocasiones a pedir o recibir ayuda.

Finalmente, intervienen otros sesgos de conocimiento conocidos como prejuicios. Estos prejuicios se desarrollan junto a otras resistencias en el Apéndice A.4.4. "El condicionamiento de las *interfases*".

### 1.3.2. El analista que no analiza, el ingeniero que no ingenia

Tras varios años de paciencia y estudio, muchos estudiantes logran constituirse en egresados. Sin embargo, y las entrevistas luego confirmarán estas palabras, éstos en ocasiones comentan: "si pudiera volver en el tiempo, probablemente escogería otra cosa". Es fácil detectarlos: rehuyen la tecnología o rehuyen las nuevas propuestas (la innovación, el cambio). Como usuarios, que pierden la paciencia fácilmente, utilizan aplicaciones diseñadas para legos. Algunos de ellos, a pesar de contar con un título en sistemas, mandan la computadora al servicio técnico cada vez que se rompe.

No todo está perdido; ya habíamos señalado que estos sujetos tienden a ser excelentes oficinistas, jefes de área y proyectistas. Aquellos que tienen paciencia, son exigentes y detallistas, se manifiestan como excelentes QA<sup>16</sup>, *Product Owners*<sup>17</sup> o analistas de formatos de APIs<sup>18</sup> de bancos —por mencionar algunas de las muchas tareas que puede cumplir alguien que tenga títulos habilitantes.

Sin embargo hay algo que queda roto: haber dedicado la vida a algo que no se siente propio. Los empleos "sustitutos" mencionados, en el mejor de los casos tienden a carecer de creación. Esta sensación es frecuente en quien se siente siempre como extranjero en su propia tierra.

16 Quality Assurance: la persona que controla la calidad de cada versión de un producto de *software* o *hardware* antes de salir al mercado, o en la jerga, "a producción".

17 En metodología de desarrollo de *software Scrum*, es la persona que traduce las necesidades de los usuarios al equipo de desarrollo.

18 API: Interfaz de Programación de Aplicaciones. En el ejemplo que nos compete: los bancos y las pasarelas de pago (Pago Facil, Rapipago, etc) publican la forma de interactuar remotamente con ellos, de acuerdo a formatos y diccionarios de datos que se envían directamente al corazón del Datacenter.

En ocasiones los egresados logran integrar su profesión con su verdadera pasión (vocación). Gran parte del *software* reinante fue creado gracias a profesionales en informática que se dedican desde componer música a crear sitios web de recetas de cocina; desde programar simuladores de conducción de trenes, a simulaciones de caída libre en el espacio, por mencionar algunos casos. Todas estas variaciones de la profesión son aceptables en tanto la persona se sienta valorada, su potencial aprovechado y, especialmente, dignifique la profesión, sin vender una imagen de lo que dice su título a las personas para luego hacerles perder tiempo y dinero.

La razón de esta declaración estriba en el gran problema que arrostra la profesión respecto de varios prejuicios instalados por informáticos que hacen ver a todos los demás como descuidados, aprovechados, poco serios o merecedores de confianza. Con estos prejuicios también deben convivir —ya sin posibilidad de retorno— abogados, políticos y obstetras.

### **1.3.3. Semblanza del egresado devenido en docente**

Es posible (y deseable) que el egresado encuentre por fin el camino hacia su propio proyecto de vida. Es usual que muchas personas aspiren a los 30 años o más a encontrar su lugar en el mundo, disfrutar del trabajo, ser reconocidos por su esfuerzo y que se les pague por ello. No obstante, es un privilegio al que pocos seres humanos arriban. De darse esta situación, seguidamente se entra en un ciclo donde se percibe con naturalidad, la complejidad y las novedades propias del rubro al que ha dedicado su vida. De haber llegado a esta etapa, en algunos casos se busca emitir una proyección del saber y de las técnicas acumuladas, a saber: el de la innovación, el de la transferencia al medio (enseñanza) y el de reajustes en función de estructuras consideradas como dignas de recuperación y puestas en valor, desestimando las innovaciones superfluas y efímeras. Este es el estado al

que se espera que haya llegado todo docente: el estado de "maestro", en su acepción de persona extremadamente versada en ciertas habilidades.

Contrariamente a esa progresión ideal, emergen otras no tan esperanzadoras y optimistas, donde no siempre el profesional llega al aula con esas condiciones ideales. Si alguien quiere dedicarse a la docencia en el área de Tecnología o Informática en el sistema educativo actual, se encontrará con que no hay concursos de antecedentes y oposición, sino un sistema de puntaje no siempre bien implementado. No hay entrevista o examen que avale la capacidad técnica del docente o, inversamente, la capacidad pedagógica del técnico, analista, programador, licenciado o ingeniero. No hay organismo colegiado —al menos en Mendoza—, que avalen una trayectoria profesional destacada. No hay que confundir profesores de informática con informáticos que hacen de profesores (es decir, puesto que cualquiera con título habilitante, puede ejercer la docencia, son "docentes de informática"). Y finalmente, se debería examinar en base a nuevas investigaciones longitudinales los efectos de posibles liviandades en los exámenes psicológicos habilitantes a docentes.

## **Capítulo 2: Resistencia al cambio**

### **2.1. Teorías asociadas al cambio**

Durante este capítulo se abordará el marco teórico relativo a la resistencia al cambio a partir de la Teoría General de Sistemas [TGS]. Se rescatará las deficiencias en el tratamiento de la TGS en las carreras de los futuros docentes de Tecnología e Informática, ya que se trata, precisamente, de una de las teorías que permiten abordar la complejidad. La propuesta es generar una discusión respecto de si el docente de tecnología es "capaz de generar cambio por sí mismo, o (más bien) ha quedado captado dentro de un juego sin fin" (Paul Watzlawick, John Weakland y Richard Fisch, 1992, pp. 111). El tratamiento del capítulo pretende esbozar una explicación mediante una introducción a las teorías asociadas al cambio, las condiciones de adaptabilidad y el cambio cognitivo. Estas últimas, medidas en términos de plasticidad mental y capacidad de innovación, y a una revisión de deficiencias existentes en el abordaje de la complejidad, ocurridas durante las carreras de sistemas, las cuales en lugar de ir a la teoría de base, se limitan a paliar el déficit mediante la inclusión de matemáticas, lo cual no es del todo errónea por cuanto la exigencia de trascender objetos abstractos de estudio, y a traer unos pocos casos interesantes de estudio como algoritmos de árboles de decisión, encriptación, recursividad o búsqueda binaria.

## 2.2. Adaptabilidad y cambio cognitivo: la plasticidad mental y la innovación

Se ha descrito al docente en varias de sus situaciones, respecto de cómo ha llegado al lugar donde está y cuál es su sentimiento. Se buscará llegar ahora al epicentro del problema, y ver si se puede trabajar desde allí.

A través de diversas observaciones en clase, y contrastadas con varias líneas de pensamiento equivalentes, existen razones para suponer que en cuanto a tecnología educativa se refiere, la ZDP (Zona de Desarrollo Próximo) tiende a reproducir matrices de enseñanza *afectadas* por factores de resistencia al cambio, de carácter afectivo durante el proceso de transmisión<sup>19</sup>.

Este carácter afectivo provoca una displicencia, una falta evidente de revisión respecto de las condiciones originales de producción del conocimiento previo, una liviandad epistemológica que Bruner (2004, pp. 104) señala de la siguiente manera:

"[...] nosotros no comenzamos con algo absoluto o anterior a otro razonamiento, sino que comenzamos con los tipos de construcción que culminan en la creación de mundos. Y estas construcciones tienen en común que toman algunas premisas como supuestas".

Cada generación de docentes de tecnología toma un legado de la precedente generación, con la potestad potencial de introducirle variaciones tanto meliorativas como peyorativas. No obstante, estas variaciones parecen ser usadas exactamente de la misma manera que en la instancia precedente. Es decir, el ser

19 Este tema ha sido ya desarrollado por Stephen D. Krashen en su teoría de los filtros afectivos en su libro *The Input Hypothesis: issues and implications*.

Krashen, S. (1985). *The Input Hypothesis: Issues and Implications*. Londres; New York: Longman.

humano inseguro, cuando decide heredar un mundo posible para construir el propio, en lugar de escoger unos cuantos axiomas seguros, usa todo el conjunto, y como un dogma ortodoxo, se cuida de incluirle variaciones que pudieran ser incompatibles o invalidantes respecto a los métodos utilizados por sus pares.

Esto supone un problema; puesto que los contextos tecnológicos cambian, es menester generar innovaciones tales que regeneren la matriz misma de los saberes previos, previendo de esta manera una involución cualitativa.

### **2.3. Deficiencias en el abordaje de la complejidad durante las carreras de sistemas**

En ambos elementos —cambio y complejidad— hay factores de resistencia que se originan potencialmente durante la carrera de sistemas, a través de una deficiente enseñanza de la Teoría General de Sistemas (o cuando menos una completa omisión de la misma). Esta teoría forma parte de los contenidos troncales de la carrera, debido a la frecuencia con la que los egresados deben lidiar con elementos emergentes que por su naturaleza, son complejos, tomando la expresión de Carlos Reynoso (2006, pp. 78): "El requisito de la teoría es que los sistemas sean relativamente complejos (pues en entidades complejas es donde se originan los llamados emergentes)".

No obstante, en la formación en las carreras de sistemas, si bien se menciona la Teoría General de Sistemas (en adelante, TGS), no se lee ni siquiera un capítulo de los postulados planteados por Ludwig von Bertalanffy. Tampoco se leen a otros autores sistémicos o cibernéticos como es el caso del padre de la cibernética: Norbert Wiener. Esta lejanía de la formación teórica redundará en una pobreza intelectual palmaria que afecta notablemente el alcance de las tesis y

los proyectos, cuyos marcos teóricos son tan pobres como sus lecturas. No hay debates teóricos sobre ningún punto; solo hay resoluciones de problemas empíricos sobre dispositivos, *hardware* o *software*, sin posibilidad de hacer ciencia pensante del quehacer universitario. En varias instituciones incluso se aprueba la carrera con proyectos de "sistemas" (dispositivos tecnológicos de funciones acotadas o específicas) que son meros sistemas de gestión, con escasa o nula transversalidad hacia otras disciplinas y sus necesidades.

El presente escrito, además, pretende documentar un tema del que no hay mayor teorización, más que debates aislados de entusiastas, alumnos universitarios, unos pocos docentes de pensamiento holístico y, en general, autodidactas. Al respecto, se puede observar el ruido que generan revisando cualquiera de los cientos de espacios virtuales existentes en redes sociales, listas de correo y foros, ruido que de una manera desordenada ha llegado a instalar propuestas en el programa Conectar Igualdad. Estas propuestas son loables en todo sentido, pero que, por una sobrestimación de las capacidades de sus beneficiarios, no han logrado romper con la inercia instalada en los docentes de Tecnología ni con la brecha digital de los educandos.

En cuanto a las teorías que explican y reflexionan sobre la adopción de actitudes que favorezcan la adopción del cambio, obviamente no serán aquellas de corte filosófico tradicional (Aristóteles, Hegel, Marx u otros por el estilo), sino aquellas que surgieron del debate de mediados del siglo XX tanto a partir de teóricos cognitivos, como de teóricos sistémicos y cibernéticos de segunda generación como es el caso de Paul Watzlawick. Este autor realizó varios avances en su libro *El lenguaje del cambio* (1989). Posteriormente, junto a John Weakland y Richard Fisch en su obra *Cambio* (1992), propusieron la detección de estados circulares que impiden la evolución de los individuos, junto a la prescripción terapéutica para su tratamiento. Por otra parte, debido a que la naturaleza del

cambio al que se asoman diariamente los docentes de Tecnología, es tanto compleja como sistémica, se recurre a la obra *Complejidad y Caos* de Carlos Reynoso (2006), la cual aporta una lúcida enumeración de los abordajes existentes, que incluye una revisión de las epistemologías sistémicas que han sustentado una u otra teoría.

El problema de cambiar implica la necesidad de darse cuenta que hay que cambiar algo. Tanto Reynoso (2006, pp. 60) como Watzlawick (1992, pp. 61) coinciden en este punto; el primero lo introduce cuando apunta al sujeto convencido que el cambio no es necesario. En sus palabras:

"si existe una realidad, ella se encuentra tan subordinada al punto de vista del actor que la contempla, que es más provechoso postular que no existe, o que es el sujeto soberano quien la constituye conforme a su experiencia privada. El argumento viene como anillo al dedo para una conciencia política desalentada que busca motivos para el abandono de las utopías: si la realidad se esfuma, desaparece también, convenientemente, la necesidad de trabajar para cambiarla. Una vez que se establece el tópico de su irrealidad, no hay más nada de lo cual valga la pena hablar".

Watzlawick por su parte denuncia abiertamente el fenómeno. En el capítulo "Las Terribles Simplificaciones" en su libro *Cambio*, lo refiere de esta manera:

"la negación de los problemas y los ataques a aquellos que los señalan o que intentan enfrentárseles van unidos. Ya que creemos que la interacción humana supone una causalidad circular y no lineal y unidireccional (en cualquiera de sus distintos niveles: familia, organización empresarial, sistema político, etc.) no tenemos necesidad de vernos envueltos en cuestiones relativas a aquello que fue antes, si el huevo o la gallina, en cualquiera de los ejemplos que seguirán. La mencionada mezcla de negación y ataque depende de groseras simplificaciones de la complejidad de la interacción en los sistemas sociales y de modo más general, de nuestro mundo moderno, cambiante,

interdependiente y altamente complejo. Tal actitud tan sólo puede mantenerse rehusando ver la complejidad de la situación y definiendo la propia y restringida visión como una actitud real, genuina y honesta frente a la vida o como un atenerse a los hechos".

Un intento de explicación puede ser que el individuo considera que en realidad "ya está haciendo cambios". La sola complejidad del mundo cotidiano moderno le requiere adaptarse constante, es decir, realizar cambios de mayor o menor escala. Algunos de esos cambios son circunstanciales y otros, que son los verdaderamente desestructurantes, que requieren una revisión de principios y de las lógicas operantes (complejidad).

Esta explicación solo puede darse si distinguimos cuanto cambio es necesario, respecto de cuanta complejidad estamos considerando. Aquí aparece por primera vez una escala para ambos conceptos: de acuerdo al grado de bloqueo o de estado circular aparente. Y, la referencia a "aparente" se debe a que todo estado de circularidad "aparentemente" irresoluble es tan solo un *cul de sac*, una rotonda que posee (al menos) una salida, incluso volviendo hacia atrás. Watzlawick ilustra el proceso de salida, dividiendo el problema del cambio en dos niveles.

Según este destacado terapeuta, hay un nivel 1 de capacidad de cambio, de tipo cotidiano, casi doméstico. El mecanismo de resolución que se emplea es de un tipo bien conocido en sistemas como "Refuerzos de tipo Positivo"<sup>20</sup>. Se ejemplifica con actos volitivos que se adscriben férreamente a la teoría aceptada. El descubrimiento de Plutón, un planeta que "debería estar allí", a pesar que no se pudiera ver con los telescopios de su época, es un buen ejemplo de algo que solo existía en la mesa de cálculo, y que solo tras una ardua labor pudo mostrarse al

20 O'Connor y McDermott (1998, pp. 57), también lo incluyen, llamándolo más pragmáticamente como "Realimentación de Refuerzo".

mundo. De la misma manera, un estudiante es capaz de obtener su título universitario oponiendo tenacidad, voluntad e insistencia antes que incorporación verdadera del sentido de los contenidos<sup>21</sup>.

Este tipo de refuerzo es el único que se emplea ante ciertas situaciones, y puede bajo ciertas condiciones provocar desestabilizaciones que veremos más adelante en Teoría de Catástrofes. Por tal razón, ante condiciones especiales de complejidad, de emergentes inesperados, de estados de bloqueo y de bucle, Watzlawick agrega la necesidad de realizar cambios de nivel 2, para lo cual introduce la *realimentación de compensación* (término más apropiado, acuñado por O'Connor y McDermott, 1998, pp. 64), y conocido en la TGS como retroalimentación negativa. Estos mecanismos representan la atenuación, freno o apertura lateral respecto de procesos que en su insistencia pueden caer en una incompatibilidad con sus propósitos de supervivencia. Ejemplos sobre este tema pueden encontrarse abundantemente en la obra de Ashby y de Foerster. Watzlawick refiere al problema varias veces en *Cambio* y entre sus contribuciones se destaca lo descabellado que parece en un primer momento oponerse o salirse de lo establecido. En sus palabras (pp. 107):

"Mientras que el cambio 1 parece basarse siempre en el sentido común (así, por ejemplo, en la receta de «más de lo mismo»), el cambio 2 aparece habitualmente como extraño, inesperado y desatinado; se trata de un elemento desconcertante, paradójico del proceso de cambio"

y no duda en recurrir a Heráclito (pp. 30), quien a su parecer

"enfocó el cambio desde una perspectiva distinta. Además de su bien conocida sentencia acerca de la imposibilidad de sumergirse por dos veces en el mismo

21 Este es probablemente el caso más frecuente de los egresados y una interesante nueva línea de investigación.

río, afirma en otro fragmento: «Todo cambio es contradictorio; por tanto, la contradicción es la auténtica esencia de la realidad»".

De esta manera, la contradicción conlleva el dolor del desarraigo de ideas ya internalizadas. Entre enfrentarse con uno mismo y enfrentarse a los demás, en general el sujeto elegirá esto último. Continúa Watzlawick explicando que

"los cambios de nivel 2 posee siempre la índole de una discontinuidad o de un salto lógico [...] por lo tanto podemos esperar que llevados a un terreno práctico (los cambios) aparezcan como ilógicos y paradójicos (pp. 32).

O'Connor y McDermott por su parte introducen sus ideas sobre el tratamiento de la complejidad usando una visión extremadamente pragmática de la TGS, señalando que la realimentación de refuerzo por si sola no es suficiente para todos los casos. Un ejemplo al que recurren es la complejidad, entrevista como una grilla cuyos nodos se encuentran conectados. La naturaleza de esas conexiones es elástica. La propiedad del conjunto es tanto más resistente en tanto aumenten las conexiones internas entre los nodos. De ahí a que si se intenta retirar o convencer uno de los nodos, será un esfuerzo fatuo puesto que todos los otros nodos ejercen un esfuerzo contrario.

Este concepto y sus propuestas de solución serían de extremo provecho para los estudiantes de sistemas, quienes deben introducir cambios y convencer tanto a máquinas como a personas. Se puede mencionar conceptos extraídos de la TGS, tales como el efecto de palanca, modelos mentales, causa y efecto, estados atractores, emergentes, oscilaciones, circuitos y tantos otros propuestos por Watzlawick, tales como manejo (o inversión) de la resistencia, menos de lo mismo, práctica del cambio, y técnicas que figuran a lo largo de toda su obra.

De lo aquí expresado puede trascenderse que los cambios de nivel primario son adecuados para enfrentar la complejidad en tanto ésta se mantenga a un nivel

comparativamente simple, de índole estática, con pocas conexiones internas. Cuando la complejidad adquiere niveles de dinamismo en el tiempo y no puede abarcarse todos los emergentes productos de las combinaciones de sus partes, es cuando debe oponerse otro tipo de esfuerzos. La Teoría de Sistemas es rica en métodos para tratarla y llevarla al necesario nivel 2 que plantea Watzlawick, donde si no puede descomponerse, al menos se puede tratar. La ausencia de estos métodos en las carreras de sistemas se nota cuando sus egresados intentan de resolver problemas aplicando las mismas soluciones que los llevaron a ese problema. Llevado al terreno psicológico, (pp. 104-5)

"en esta posibilidad de ser diferentes las "realidades" subjetivas (realidades del segundo orden) radica el poder de las intervenciones terapéuticas conocidas como reestructuraciones".

Watzlawick demostró en sus terapias que el cambio era posible y así lo señaló en *El Lenguaje del Cambio* (1989, pp. 105), cuando refirió a que

"nunca nos enfrentamos con la realidad en sí, sino sólo con imágenes o concepciones de la realidad, es decir, con interpretaciones. El número de posibles interpretaciones para cada caso es muy grande. Pero, en virtud de la concepción del mundo del interesado, suele casi siempre quedar reducido, a nivel subjetivo, a una que parece ser la única posible, razonable y permitida. En razón de esta única interpretación, la mayoría de las veces sólo se considera posible, razonable o permitida una sola solución, y cuando esta solución no lleva a la meta apetecida, se busca, típicamente, más de lo mismo. Aquí es donde entra en función la reestructuración, y con excelentes resultados, a condición de que se consiga prestar a una determinada situación un sentido nuevo, también adecuado o incluso más convincente del que le ha venido dando hasta ahora el paciente".

## 2.4. Mecánica en lugar de sistemas

Es llamativa la omisión en materias como Sistemas de Información o en Sistemas de Procesamiento de Datos respecto de La Teoría General de los Sistemas, considerando conceptualmente lo cerca que ésta se encuentra de los modelos que usan informáticos y tecnólogos. Una vez que se estudian los elementos que conforman esta teoría, y se sigue cualquiera de sus implicancias en el terreno cognitivo, se detecta que su incorporación es esencial debido a que la complejidad es inherente a los tipos de problemas que suceden en los sistemas.

Tal como se imparte actualmente la Teoría de Sistemas en estas carreras, no sirve para brindar herramientas que permitan abarcar la complejidad. Para el estudiante de primer año, la Teoría de Sistemas es apenas el boceto de una caja negra que posee una entrada, un proceso interno y una salida. Todas las demás propuestas de la teoría son desechadas, en función de continuar la explicación centrada en el proceso interno, explicado con álgebra de Boole; conjuntos de compuertas que realizan sumas y productos binarios, para pasar a la CPU y otros circuitos.

Lo extraño del caso es que basta mencionar a la TGS durante el ejemplo mencionado en esta sola clase, para agregar "Sistemas" a la denominación de la carrera. Un epistemólogo poco avezado creería que los Analistas e Ingenieros en Sistemas son expertos en tratar la complejidad, cuando en realidad tienden a ser mecanicistas reductores, preparados para actuar sobre modelos de tipos deterministas, canónicos, estadísticos y expertos (Reynoso, 2006, pp. 28).

Esta aseveración no contiene tintes peyorativos: los modelos mecánicos sirven para explicar muchas cosas, como la Ley de Gravitación Universal (2006, pp. 28), cadenas de Markov, camino estocástico, agregación limitada por difusión, movimiento browniano (Reynoso 2006, pp. 32), e incluso termodinámica (Reynoso

2006, p 33). Acaso intuitivamente, cuando en informática se adosa tiempo, interacciones y elementos de azar al modelo mecánico, se obtienen comportamientos emergentes como autómatas celulares o criptografía, utilizada en autenticación de usuarios (ejemplo mío). Acaso como paliativo por la falta de formación en complejidad, es que se agrega matemática al cursado, y ni siquiera ésta conectada con ejemplos reales como las implicancias de los algoritmos de predicción semánticos de Google, o del recorte de la distancia geodésica medida en *hops* de contactos en redes sociales.

¿He reducido lo suficiente? ¿Estoy tratando algo de forma irreductible algo que debería seguir indagando? ¿He llegado a la raíz del problema, o solo lo estoy aplazando? Estas son preguntas que debería hacerse el informático cada vez que asegura que ha resuelto algo ante los neófitos.

Por esta razón, aprender cuando y cuanto sustraerse para abarcar los sistemas desde afuera (visión holística), y por el contrario, inmiscuirse y entender el comportamiento del todo mediante la explicación de sus partes (visión reductora), es una actividad que los alumnos deben aprender de forma intuitiva *después*, en sus años profesionales, cuando bastaría tratar adecuadamente el tema al principio de sus carreras de sistemas. Los egresados deben llegar a estas conclusiones por sí solos, de una manera no guiada que hace que pierdan la conexión con sustratos enriquecidos por propuestas, al menos referencias en clase, de otras opciones como la Teoría de las Estructuras Disipativas (o de los Sistemas Alejados del Equilibrio) promovida por el Premio Nobel Ilya Prigogine desde principios de la década de 1960, continuada por la Cibernética de Segundo Orden de Heinz von Foerster, la Cibernética Conversacional de Gordon Pask, la Autopoiesis de Humberto Maturana, la Enacción y la Neurofenomenología de Francisco Varela, el Constructivismo de Ernst von Glasersfeld o la Teoría de Catástrofes de René Thom, elaborada hacia 1970.

Por supuesto, esta es una carga inmensa para incluir en el programa de estudios de las carreras de sistemas. Pero de todas ellas basta con seleccionar y tratar sus principios generales de Totalidad, Homeostasis, Equifinalidad, Multifinalidad, Morfogénesis y Jerarquía propuestos por la TGS (von Bertalanffy, 2003, pp. 54 y ss., 136; Reynoso 2006, pp. 78-84). Esto permitiría aproximarse al estudio de los sistemas de una manera más completa, realizando síntesis más integradoras, sin que ello implique forzosamente perder complejidad (Watzlawick, 1989, pp. 43). Es decir, sin caer en reduccionismos mal hechos, quitando partes que no deberían haberse quitado, tales que desdibujen en sus alteraciones simplificadoras.

Llevado a un terreno áulico y a un ejemplo observado mientras se realizaban tareas de encargado de laboratorio, se pudo detectar que los docentes de informática tienden a señalar como problema de Linux que una impresora no funcione, sin explicar que los módulos o "*drivers*" deben ser abiertos antes de ser aceptados en el *kernel*. Y que esto es un problema para algunas compañías de *hardware*, que esconden ciertos procesos en los *drivers* que deberían ser ejecutados por la impresora, no la CPU, con el fin de ahorrarse dinero. Obviar esta explicación es muy grave: si un alumno entiende que "no puede imprimir con Linux", entonces no usará este sistema operativo para ninguna otra tarea. En tal sentido una simple investigación por parte del docente le demostraría que las empresas que fabrican impresoras, también proveen archivos PPD en lugar de *drivers* para estos casos. Incluso si hace honor a su profesión, y a la manera de Watzlawick, debería resolver el problema de fondo: proponer no comprar estas impresoras, boicotear marcas deshonestas que venden *hardware* "defectuoso de fábrica". Munido de los varios niveles de programación que aprobó en la universidad, puede crear *drivers* libres mediante ingeniería inversa, o resolver el tema como se hizo hasta mediados de los años 90, escogiendo el *hardware* en

función del *software*, mediante consulta previa en listados llamados *hardware compatibility list*.

Tras lo expuesto, es preciso indicar que no hace falta llegar al estudio del debate entre Heráclito y Parménides, o de estudiar completamente la TGS para incorporar la gimnasia de desconfiar de las respuestas simples y de indagar lo que hay más adentro: hay resúmenes y enfoques modernos, desarrollados por autores contemporáneos en torno a la teoría de sistemas, tales como Joseph O'Connor y Ian McDermott (1998) quienes sugieren toda clase de técnicas al respecto. Con lo cual, el adecuado tratamiento de la bibliografía durante la carrera es una posible respuesta al problema del tratamiento de la complejidad dentro el ambiente de sistemas y su enseñanza.

El punto se comprueba si se revisan los conceptos propuestos por estos últimos autores. Estos conceptos son ni más ni menos que un resumen del pensamiento sistémico tal cual fue propuesto por los padres de la cibernética actual: estabilidad, efecto de palanca, pensamiento circular, realimentación de refuerzo, de compensación, modelos mentales, caras del caos, autoreferencia, y recursividad entre otros. Estos elementos se pueden considerar transversalmente, en un marco de generalidad tal que trasciende las máquinas donde fueron aplicadas, y que aportados a los estudiantes de sistemas, pueden prestarles una valiosa ayuda para diversas situaciones en su vida profesional. De esta manera, si un informático desarrolla un tema complejo frente a los alumnos, debe inmediatamente reflexionar si lo está reduciendo correctamente. De tener la gimnasia que aporta la TGS, entendería que no se puede subestimar un problema complejo dando siempre una respuesta simple, porque para empezar, una de las leyes de la complejidad es que ésta nunca se explica monocausalmente. Así, tampoco podrá alegar como excusa la inconveniencia de la precisión o de la necesidad momentánea ("en clase") de cercenar partes, sin tomarse la molestia

siquiera de incluirlas en los apuntes. Así como "Dios está en los detalles", persistir en una respuesta sesgada en aras de evitarse la conservación de energía psíquica que requiere su correcto tratamiento, es un exceso ciego de realimentación positiva que la TGS distingue muy bien.

## Capítulo 3: El Lobo Solitario

En el argot informático, el lobo solitario o esteparo —*alone wolf*<sup>22</sup>— es aquel individuo que resuelve las cosas a su manera, sin pedir ayuda a nadie, a semejanza de los vaqueros de las películas. Muchos *alone wolves*<sup>23</sup> son considerados geniales —y la mayoría lo son—, pero aún más por lo mucho que dejan al imaginario de los legos: "si no pide ayuda, es que no la necesita".

En ocasiones, el docente de tecnología argentino manifiesta un comportamiento sociopático y competitivo parecido respecto de sus colegas, y especialmente ante los alumnos. Sin embargo, la falta de comunicación no esconde otra cosa que una actitud sapiencialmente menesterosa, competitiva y enmascaradora de las deficiencias cognitivas y operacionales. Es un síndrome del *alone wolf*.

El comportamiento es previsible y hasta esperable de informáticos concentrados en complicadas configuraciones, pero no de docentes, cuya principal virtud es acaso la comunicación. Si un *alone wolf* se ve obligado a recurrir a la docencia como refugio económico (o como refugio de la complejidad), y no es capaz de aperturarse hacia alumnos y colegas, sus efectos tendrán un impacto condicionante tanto para los alumnos (Watzlawick 1989, pp. 44) como para sí mismos (Reynoso, 2006, pp. 96). Hacer una revisión de los esclerosamientos en los métodos y en los contenidos (Watzlawick, 1989, pp. 104-5) resulta vital tanto en un ambiente dinámico como es el de la Tecnología, ya que tarde o temprano los supuestos se desplazan, se pone en evidencia la falta intencional de aperturidad y el énfasis en la simplificación de los argumentos (Watzlawick et al., 1992, pp. 61-2). De esta manera, los emergentes, en ocasiones, asumen la

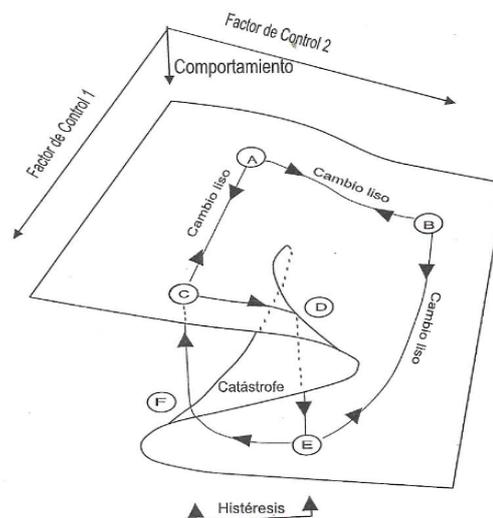
22 El concepto de lobo solitario (*alone wolf*), tomado del alemán *der Steppenwolf* ha sido tomado de la celeberrima obra de Hermann Hesse.

23 El sustantivo *wolves* es el plural de *wolf*.

posición de paradigmas hegemónicos, y el docente se ubica cada vez en una situación más grave (Watzlawick et al, 1992, pp. 68), tanto más si continúa o se afianza en posturas anquilosadas y superadas en otros ámbitos, pudiendo caer en algo bien definido por otra de las teorías que tratan la complejidad, la Teoría de Catástrofes de René Thom, una situación de caída llamada Catástrofe de Cúspide (Reynoso, 2006, pp. 167), donde pueden encontrarse también elementos de la Teoría de Sistemas: multifinalidad, equifinalidad, estado estable y morfogénesis.

La propuesta aquí, y a guisa de mostrar la manera en que estas teorías sistémicas pueden ayudar a los profesionales de sistemas, consiste en emplear algunas de las terapias propuestas por Watzlawick, para proponer un proceso inverso de recuperación desde "adentro" del pliegue hacia "afuera", mediante un fenómeno especial llamado

*histéresis*. Esta consiste en la capacidad de un objeto o fenómeno (natural o social) de mantenerse en la misma dirección hasta que el mismo estímulo original ausente lo vuelve a invertir: el *cul de sac* expuesto en el capítulo anterior. La histéresis se encuentra asociada a muchos procesos dinámicos, desde los circuitos electrónicos hasta las psicosis maníaco-depresivas (Reynoso, 2006, pp.



*Ilustración E: Superficie de Riemann-Hugoniot, usada como Catástrofe de Cúspide por Zeeman E (1976, pp. 65-83)*

176). Entre las variables a considerar como histeréticas —según Watzlawick<sup>24</sup>— cabe mencionar la comunicación con sus pares (factor negentrópico o disruptivo que genera un quiebre con respecto a las matrices de acción precursivas hasta dicho momento) y situarla como Factor de Control 1; ello se podría vincular con una escala respecto de la cantidad de complejidad que puede abarcar, como la que se plantea más adelante en el Análisis de Datos, en el Factor de Control 2. Con base en estos parámetros se puede construir un modelo predictivo como el de catástrofe de Cúspide —cfr. *supra*, ilustración E.

En caso que se quisiera plantear una versión más sofisticada de esta terapia, se puede trabajar a partir de modelos de tres factores, siendo el tercero de ellos una variable utilizada más adelante durante el análisis de datos, denominada *componente motivacional*. De allí se avanzaría a catástrofes tales como la Cola de Milano, el Ombligo Hiperbólico o el Ombligo Elíptico (estos últimos con dos desenlaces posibles).

Finalmente, a fin de satisfacer la necesidad imperiosa de "apertura" que menciona Watzlawick, también se puede proponer como variable de control extra la *capacidad de realizar cambios transversales*. Al respecto varias vías de resolución se brindaron en el ejemplo de la impresora, en el capítulo precedente. De esta manera se pueden conformar cuatro ejes, tales que lleven esta propuesta a una versión tetradimensional, computable en modelos como la Catástrofe de Mariposa o el Ombligo Parabólico (uno y dos desenlaces posibles, respectivamente).

Este estudio hipotético requiere de varias muestras en el tiempo (investigación longitudinal), y no se propondrá más que como ejercicio al lector, en tanto que durante el Análisis de Datos se trabajará con metodologías ya maduras

24 Las técnicas mencionadas pueden examinarse en el Apéndice C: El pensamiento estratégico necesario en las carreras tecnológicas.

como estudio de casos y análisis comparativo desde la perspectiva de Jordi Caïs (1997).

### **3.1. Impacto de la resistencia al cambio en el docente:**

#### **3.1.1. Cese de investigación**

Obtenido el título habilitante, el informático devenido en docente de tecnología cesa el ritmo de investigación propio de la facultad, y establece prontamente un sistema de creencias como verdades absolutas incuestionables. Aquello establecido permanece como un credo secular ortodoxo, en el que las cosas se deben dejar tal cual están, para que no se "rompan" y cuyas ortodoxias se autojustifican de manera típica al sostener que "son los métodos que han permitido la existencia de todo lo actual"<sup>25</sup>. Ello se considera en el área de sistemas como un movimiento de realimentación por refuerzo, que en un punto se ha desbocado por carecer de un contra mecanismo de compensación (O'Connor, McDermott, 1998, pp. 58 y 64).

Esta lógica se contrapone a la lógica científicista, que incluye experimentación al quehacer de lo cotidiano, teniendo como premisa que la base

25 Sin embargo, aquellos sistemas que se encuentran en equilibrio lo son sólo porque encontraron un nicho en la escala evolutiva que les permite sobrevivir sin problemas, pero que muchas veces representan también un atasco recesivo en el sentido lakatosiano. Por emplear un ejemplo transversal a biología, si bien puede otorgarse mérito evolutivo a una bacteria extremófila que procesa azufre en una fumarola volcánica submarina, no quiere decir que también pueda adjudicárseles una capacidad mayor de supervivencia. En efecto, si las condiciones ambientales se alteran en lo más mínimo, puede ocurrir que la bacteria colapse. Ese riesgo se encuentra a unos 15 centímetros de la chimenea volcánica, y es la razón por la que deba esporularse en segundos para resistir las condiciones adversas del entorno no extremófilo.

del conocimiento está fuertemente asociada a la desconfianza de todo aquello que no se encuentre constatado empíricamente, o que al menos difiera de líneas de razonamiento deductivos consensuadas con la comunidad científica. El sujeto que experimenta, luego de una observación de un objeto en estado de homeostasis, normalmente debe alterarlo ("descentrarlo") para aprender de él. Ejemplos sobran y van desde un biólogo estimulando un protozoo, a un físico interfiriendo sobre la trayectoria de una partícula. Frecuentemente *deben* descentrar completamente a sus objetos rompiendo sus homeostasis, equilibrio o trayectoria, según el caso. No siempre el caos (otro concepto que no se estudia en la carrera) es inútil o indeseable. De hecho, si se le pregunta a un biólogo o a un genetista, se podrá comprobar que el caos es necesario en términos evolutivos. No obstante, este "descentrado" de la estabilidad es costoso. Los físicos por ejemplo, saben que pueden llevarse desagradables sorpresas cuando rompen átomos.

Sin embargo, en el caso de los docentes de tecnología e informática, es curiosa la falta de investigación en su campo. Como si para probar software nuevo tuvieran que construir un acelerador de partículas. Por ejemplo, es muy preocupante que eviten el simple, barato e interesante experimento de formatear su unidad de disco e instalar otra cosa que no sea lo que usan todos los días. Tan sólo experimentar con otras herramientas, respecto de las que usan habitualmente, los pondría ante el descubrimiento de novedosas líneas de pensamiento, propuestas por instituciones educativas o fundaciones sin fines de lucro, en lugar de corporaciones. *Linux* (Universidad de Helsinki), *FreeBSD* (Universidad de Berkeley), *Minix* (Vrije Universiteit, Ámsterdam) o *Haiku* proponen razonamientos completamente originales, desde el momento que fueron abiertos al mundo.

De esta manera, el "laboratorio" de experimentación del informático es tan accesible como sus propias máquinas hogareñas. Sergio de Regules (2016) ilustra

la importancia que significó para Edward Lorenz, quien pudo detectar el efecto mariposa descritos pero nunca demostrados por Poincaré y Fontenelle, atrapados ambos por las limitaciones de su época: “El ordenador fue para Lorenz lo que el telescopio para Galileo: el instrumento esencial para ver lo que nadie había visto”. Lorenz en el año 1964 le rindió homenaje a su calculadora Royal McBee LGP-30 de apenas seis dígitos. Hoy en día es dudoso que alguien pueda enaltecer de la misma forma la ingeniería de una *netbook*. Estos equipos, aunque modestos, son miles de veces más potentes que la calculadora que empleó Lorenz para determinar nada menos que la metafísica que existe entre los conceptos “*determinista*” y “predecible”, creando por el camino una nueva rama de la ciencia, llamada Caos.

Del lado del docente de informática, lejos se encuentra su intención de vencer los límites del equipo o de realizar un examen respecto de cuanto potencial se desperdicia usándola a guisa de máquina de escribir. El temor por descontrolar lo controlado, de descomponer la herramienta, genera un pánico que inmediatamente va acompañado de un temor instintivo a todo proceso lúdico de exploración e innovación, al punto que mina todo proceso de enseñanza o de aprendizaje, tanto de ellos mismos, como de sus estudiantes, quienes simplemente siguen rutinas y procedimientos seguros, preestablecidos tanto en cuanto a lo que hacen, como en cuanto a lo que obtienen. Esto es un sesgo de toma de decisiones denominado aversión a la pérdida, según el cual el preferible *no perder* antes que *ganar algo*.

La inferencia que se puede realizar aquí es que el profesor en informática, o el informático devenido en profesor, *justifica los errores de diseño, no discute sus inconsistencias*. No hay problema, por lo tanto, *no hay necesidad de investigación*.

Watzlawick choca de frente con esta postura<sup>26</sup> al preguntar: "¿Cómo podemos tener la certidumbre de que un sistema está libre de contradicciones si nosotros mismos nos encontramos en el interior de un sistema?" (1994, pp. 348).

¿Cómo se hace para ignorar estas contradicciones cuando se presentan de modo tan patente y palmario? Cabe plantear dos posibilidades (mutuamente inclusivas). Por un lado, el profesor tiene fe en que terminará su día sin pérdida de datos; por ejemplo sostiene que *Windows* es más estable cada año, de modo que hasta puede emplearlo como justificativo para no cambiar a Linux. Conclusión a la que arriba, partiendo de una premisa falsa, un sesgo del conocimiento basado en un abuso del anclaje (*anchoring*) que no es otro que un prejuicio del *status quo*. Metafóricamente es similar a la figura, salvando las distancias, de un pueblo conducido a través de un desierto, para llegar finalmente a la tierra prometida por el departamento de *marketing* de una empresa. Por el otro lado, *el profesor posee tal respeto por el sistema hegemónico actual que es capaz de realizar ciertas concesiones, tales como sacrificar sus datos y el de otras personas, mientras no se desequilibre el (des)equilibrio existente*. Cabe plantearse entonces, si en esto se parece a los usuarios, si su consejo es *no cambiar lo que está funcionando, solo usarlo tal cual está*, ¿para qué se lo está contratando? Luego, durante el corolario, se volverá este tema, pero se entiende aquí que su figura y de su espacio aúlico, repleto de *softwares* cada vez más *fáciles* de usar, lo ponen en una situación donde nadie lo va a necesitar en un futuro cercano.

26 Hay varias citas de Paul Watzlawick en la presente obra. Esta en particular, puede encontrarse en la entrevista realizada por Carol Wilder para el libro "La Nueva Comunicación", referenciada en la sección de Bibliografía.

### 3.1.2. Chauvinismo del idioma materno

Una manera simple de observar la falta de adaptación en el docente de tecnología consiste en revisar el lenguaje que usa para su trabajo diario. Como hispano hablante nativo, el español, con todo el aprecio y el disfrute que me supone usarlo, es limitante desde el momento en que es usado sólo por el 17% de los usuarios de Internet.

Otro aspecto al que el docente se pliega incondicionalmente, es al exceso que se hace en la sociedad argentina respecto del conservacionismo de nuestra lengua, como si fuera a extinguirse. Es cierto que a veces la jerga tecnofílica adopta cualquier palabra en inglés como si no existiera un correlato igual —o hasta más claro— en la propia lengua madre, lo cual constituye una transculturación básica. Sin embargo nada se dice de los políglotas europeos, quienes por necesidad, deben incorporar varias lenguas. Esto como consecuencia forzada de su propio pasado cercado, medieval, chauvinista, que como veremos a continuación, es una tendencia actual de nuestro propio país.

La pseudoprotección de la cultura autóctona respecto del lenguaje inglés proviene de torpes acciones que buscan paliar tanto nuestra histórica inhabilidad comercial en comparación con la inglesa, como de esconder nuestro conformismo ancestral. Convengamos que hay algo esquizofrénico en utilizar en español la *interfase* de *Facebook*, para poder luego "postear" inflamadas proclamas anti imperialistas. O como apuntara Wolton "uno de los efectos de la dominación cultural es, precisamente, no pedir otra cosa que lo que se tiene. Desear otra cosa, emprender, ya es situarse en un límite dinámico de cuestionamiento, de emancipación" (2000, pp. 107).

Esta esquizofrenia se observa en diversas acciones gubernamentales para aislar el país del mundo. En medio de una crisis global de alimentos, Argentina se

prepara para cerrarse a los "bárbaros", tal como lo hicieran los pequeños estados del medioevo, cuyas consecuencias actuales se observan fácilmente por la cantidad de dialectos resultantes que germinaron tras las murallas.

Desde hace varios años se observan acciones de ideas políticas del siglo pasado que se escudan en un pseudochauvinismo respecto de una urgente protección de la lengua local: el castellano, como si ésta fuera una lengua en extinción. Esta situación se agrava aún más ante la ley que obliga a la traducción del audio de las películas extranjeras, perdiéndose así el residuo cognitivo que supone la dualidad de voces y subtítulos en simultáneo, y que permite la adquisición de una segunda lengua para aquellos que desean comparar la traducción realizada por otros con lo que escuchan.

### *3.1.2.1. Breve apología por el lenguaje universal de facto*

Si bien es correcto traducir toda expresión mientras ésta no pierda su sentido, en tecnología esta posición se encuentra todo el tiempo en un punto ambiguo. Existen muchas razones para emplear el lenguaje inglés en tecnología, cuyo desmerecimiento supone dejar pasar aquellas pocas ventajas que pueda poseer la globalización para el bloque regional.

Por supuesto dejar pasar una lengua sin más a un territorio requiere de una discusión de fondo. Este texto es consciente del hecho y del peligro de la transculturización que sigue a la instalación de un idioma por *de facto*. Pero también trata de ser contemplativo respecto de sus beneficios en función del tiempo. No es la primer vez que pasa en la historia. Podemos retrotraernos al Lejano Oriente durante los siglos II a.C. al II d.C. Durante la dinastía Han con el emperador Amarillo (Huang Ti), quien buscó unificar a los distintos reinos y pueblos que comprendían su imperio. Este emperador hizo destruir todo texto que

no estuviera escrito con los 80.000 ideogramas que serían los únicos válidos para escribir cualquier texto de su imperio. Su unificación arrasó violentamente con múltiples tradiciones y raíces culturales milenarias. Sin tanta violencia de por medio, en la India se impuso como lengua sagrada el devanagari o lengua prakrita (el sánscrito) al cual se vertieron los textos más antiguos de los Vedas, el Vedanga, los grandes Itihasas, los comentarios de los Puranas y Brahmanas, así como más tarde se hará con los textos del budismo mahayana, que llegarán incluso al antiguo Tíbet, Nepal, Bhutan, China y Japón.

Otro fenómeno semejante ocurrió en Cercano Oriente con la hegemonía de la cultura egipcia, que durante cerca de 2.500 años fue la lengua de la cultura, durante la cual se pueden advertir a lo sumo 5 grandes variaciones en su escritura producto de crisis y cambios sociopolíticos. Más tarde, durante los siglos VII, y más específicamente entre los siglos V a.C. y II d.C. la lengua griega pasará a ser la lengua de la alta cultura y a la cual se traducían casi todos los textos de las civilizaciones mediterráneas, y que dio lugar a un nuevo giro hegemónico con la absorción de la cultura griega por la cultura romana y su lengua latina. La adhesión y triunfo de la patrística latina adoptada por el Imperio Romano y la difusión del cristianismo por toda Europa condujo a la adopción de esta lengua como la lengua sagrada de la iglesia, así como la de todo conocimiento con pretensiones de universalidad hasta entrado el siglo XVIII. Durante el siglo XV, Johannes Gutenberg introdujo un quiebre paulatino de la hegemonía de la lengua latina, como lengua universal, ya que establece la primer publicación de la Biblia en idioma alemán, instalando así una revolución que dejó de tener numerosas oleadas hasta nuestros días. Este quiebre junto con la aparición de las primeras academias de ciencias (Royal Society of London for the Promotion of Natural Knowledge, Sociedad Henry Oldenburg, Académie Royale des Sciences Française, Preußische Akademie der Wissenschaften, Königlich Preußische Sozietät der Wissenschaften, Kurfürstlich Brandenburgische Sozietät der

Wissenschaften, Deutsche Akademie der Naturforscher Leopoldina), etc. Estas academias instituyeron comunidades científicas y legitimaron sus idiomas como idiomas de la alta cultura mundial. Más tarde aparecerán academias de ciencias específicas en España pero habrá que esperar al siglo XVIII y XIX, con lo cual se puede apreciar el espíritu retardatario de nuestros antecesores en materias de introducción de la lengua castellana como lengua de la ciencia.

Si se tiene en cuenta que la lengua inglesa fue la impulsora —junto a la francesa— de la revolución industrial, así como también lo fue la colonización mercantil naval inglesa de los mercados y la ulterior globalización del mercado de hegemonía estadounidense, no es cuestión baladí que los tecnólogos contemporáneos adopten este idioma como idioma transcultural de imposición en sus discursos científicos y de difusión técnico-tecnológica. El lenguaje inglés es un idioma común instalado en la mayor parte de las naciones por *de facto*.

El hecho es que si se toma al azar a dos personas en el planeta y se calcula las posibilidades que se puedan comunicar, aumentará en grado sumo sus posibilidades si ambas al menos escriben y leen en inglés. Y no solo aumentará mucho la posibilidad de que se comuniquen, sino que se entienda y hasta coincidan en los mismos intereses. Lamentablemente no hay manera de mejorar esto ni siquiera con la tecnología existente: esta apenas puede ayudarlos en exactitud y tiempos de respuesta<sup>27</sup>. Incluso es menester que ambos interlocutores tengan una competencia lingüística muy buena en el uso de sus respectivas lenguas base para que la traducción electrónica tenga una proximidad aceptable en el texto meta. No incluyamos la traducción automática de la voz hablada,

27 La Web 3.0 semántica, que prometía la *intertraducción*, no ha sido más que una utopía. Por ejemplo, la traducción del japonés al español en tiempo real, sin pérdida de significado, es imposible con la tecnología actual. Al menos una de las dos partes debe emprender seriamente estudios del otro idioma si pretende darse a entender sin graves reduccionismos del otro lado.

porque raya en la irrelevancia por el poco avance que se ha logrado en ese sentido.

En tal sentido, es indiscutible la simplicidad de la estructura y del vocabulario de la lengua inglesa (26 letras en común con el español), frente a lenguas como la china. Ello facilita la implementación de esta lengua, y es coherente que sea "el" lenguaje de la red.

Otro factor que coadyuva la adopción del inglés como lengua de uso masivo en el campo de tecnología digital es la demanda laboral de personal que domine el inglés para ocupar puestos en el campo empresarial de sistemas, informática, telecomunicaciones y redes. La terminología en estos campos no se traduce ya a otros idiomas como se intentó realizar en las décadas de los '80 y los '90 del siglo pasado. Como ejercicio, trate el lector de expresar con la menor cantidad de términos posibles el significado de los siguientes conceptos. Como ejemplificáramos anteriormente, traduzca *IT Manager*<sup>28</sup>, *DevOp*<sup>29</sup>, *Social Media Manager*<sup>30</sup> o *User Experience (UX/UI) Specialists*<sup>31</sup>.

Tan sólo la traducción de estas nuevas profesiones arroja varios inconvenientes. Para cuando se establezca su significado en el medio actual,

28 Refiere al administrador de sistemas, que incorpora tareas de infraestructura y comunicaciones.

29 Administrador de sistemas en ambientes de *cluster* (conglomerados de servidores físicos, virtuales o contenedores).

30 Se trata de la persona que mejora la presencia de las empresas en buscadores y redes sociales, mediante el uso de reglas de organización de búsquedas inteligentes, adopción de estándares de protocolos y normas de usabilidad.

31 Programadores y diseñadores que construyen *interfases* enriquecidas, focalizadas en las experiencias de los usuarios.

probablemente hayan mutado a nuevas definiciones o directamente dichos campos hayan sufrido reconversiones significativas que las tornan obsoletas o, al menos, obsolescentes. Y puesto que sus contratantes y beneficiarios son personas del mundo, el esfuerzo por lograr su traducción se limita a los lingüistas locales, quienes complican aún más la acepción duplicando torpemente el significado o generando traducciones confusas por falta del dominio técnico del campo.

Borges y Sábato se manifiestan en forma admirablemente lúcida cuando citan sin traducir, indicando de esta manera que "siempre" hay pérdidas drásticas de riquezas semánticas durante toda traducción; incluso habiendo comprensión total del significado, se pierde el rastro de las expresiones que dieron un marco a su concepción.

Otros aspectos más a considerar tienen que ver con la posición desventajosa que tenemos, con respecto de habitantes de otros países que aceptan el idioma inglés sin mayor polémica.

El primero tiene que ver con los manuales, actualmente devenidos en tutoriales: el *know how*, el saber cómo (asociado al saber hacer), se encuentra en estos documentos paso a paso, normalmente más actualizados y revisados en inglés. Si se conecta a un foro de *Photoshop*, se podrá constatar que todos los profesionales usan sus versiones del programa en inglés, en tanto que los novatos hacen proezas para encontrar las secuencias equivalentes de instrucciones dentro de los menús de su versión en español.

El segundo aspecto bien lo conocen aquellos diseñadores y programadores de Argentina que mediante portales *freelance*, o empleados en factorías de *software*, se encuentran vendiendo progresivamente cada vez más servicios al exterior. Poca gente sabe el tremendo auge que ha tenido el trabajo por *Internet*,

al punto de convertirse en otra industria sin chimeneas. La competencia con *indos* es diaria, con la ventaja de que ellos tienen como lengua oficial el inglés y no el *hindi*.

El docente argentino se confunde y a menudo es prejuicioso, aislacionista y, nuevamente, un *alone wolf*. Si yo tengo una parte de la solución a un problema técnico y otra persona tiene la parte restante, entonces toda discusión extra resulta fatua y baladí.

Deberíamos aprovechar como hispanoparlantes, la ventaja al menos de usar el mismo juego de símbolos empleados por la lengua inglesa. Aprovechar que no tenemos la doble dificultad de enfrentar otras culturas como la rusa con sus caracteres cirílicos (кириллица), la hindi (देवनागरी) (devanagari), la العربية (árabe) o la העברעיש (hebreo) por mencionar algunas. Además, a la diferencia de los caracteres de escritura, a estas culturas se les agrega el problema de la dirección de la escritura. Tanto el árabe como el hebreo presentan una orientación de escritura de derecha a izquierda. Si usted está tentado a leer este comentario de corrido, lo invito a girar el texto (o a colgarse del techo boca abajo) y a leerlo nuevamente, para tener una idea más acertada del problema, ya no traducción, sino además del problema que conlleva interpretar los símbolos.

En comparación, estas trabas resultan nimias respecto de las dificultades de índole semántica, que nuestra lengua prácticamente carece al momento de realizar una traducción del inglés. Teóricamente, la resistencia debería reducirse considerablemente en estas latitudes. Paradójicamente nuestra resistencia es mucho mayor a la de países como India, China, Rusia o Japón, quienes tienen incluso razones extras —históricas— para defenestrarlas. Sin embargo ellos dejan el pasado de lado y no tienen tantos remilgos ni excusas como las que argumentan los profesionales que se dedican a la computación, la tecnología o

informática en Argentina. En estos países, acaso más acuciados por sus necesidades, aprovechan las ventajas de la globalización en lugar de aislarse como esquizofrénicos *alone wolves*.

### **3.1.3. El docente en la torre de marfil**

El término “Torre de Marfil” se usa desde el siglo XX para designar un mundo o atmósfera, donde los intelectuales “se involucran en actividades desconectadas de las preocupaciones prácticas de la vida cotidiana” (Gullo et al, 2013). El autor refiere también a la expresión de Kirsten J. Broadfoot (2008), quien señala que una Torre de Marfil sería una entidad de razón, racionalidad, y estructuras rígidas que colonizan el mundo de la experiencia vivida, y apunta a que el trabajo escolar se debería fundar en el diálogo, en lugar que un grupo o un individuo edifique una esencia de exclusividad y superioridad. En un sentido académico, esto conduce a un dominio desproporcionado sobre la diversidad, creatividad y cultura de la organización. En estas circunstancias, la imagen de la Torre de Marfil es peligrosa por cuanto privatiza el conocimiento y el intelecto.

Hay razones para suponer que la resistencia manifestada durante la referida migración a software libre descrita al principio de esta obra, y a la insistencia respecto de utilizar solo programas de la empresa Microsoft, se imparta desde la figura “Torre de Marfil”. Una de éstas razones estriba en que ninguno de los siete docentes afectados se hayan dignado a explicar sus razones para oponerse al cambio, incluso conminados a describirlas por escrito por Dirección. Otra de las razones se encuentra relacionada con la seguridad a la que han arribado, un estado precariamente morfoestático, en el cual consumir aplicaciones de la "nube", *bloggear*, *retwitear* o dominar los narcisistas teléfonos celulares de última generación, se presenten como acciones representativas ante los legos, que los validen como individuos que se encuentran al día con la tecnología.

Respecto de esta afirmación, si bien para un usuario es admisible la vanidad que pueda enarbolar por dominar estos dispositivos tecnológicos, el mismo comportamiento es al menos ingenuo para un profesional que posee competencias en tecnología avaladas por un título. *Confundirse* entre la generación consumidora de tecnología es, literalmente, *confundirse*, puesto que no incluye la comprensión que conlleva la innovación, advertir sus peligros ni cómo aprovechar sus ventajas. Consumir un producto no es aprovecharlo, ni usarlo todo el tiempo, equivale a amortizarlo. El docente de Tecnología debería preciarse de tales habilidades, cuando sea capaz de elaborar proyectos que tengan sentido desde su construcción, y no solo desde su mero uso. Siguiendo con el ejemplo del celular, no es difícil

- a) bajar un SDK (*Software Development Kit*) de la compañía de celulares.
- b) tomar un tutorial, crear un proyecto y pegar un poco de código fuente;
- c) Cambiar un poco los ejemplos y aprovechar:
  - cámara: capaz de detectar movimientos;
  - acelerómetro / gravitómetro;
  - GPS; y
  - conexión a Internet de la escuela.

Estos elementos constituyen componentes que pueden combinarse con las *netbooks* y con el servidor del colegio. Se los puede enlazar a algún servidor gratuito de base de datos como *MySQL* (instalado en una *netbook* o en algún *hosting* barato), y construir pequeñas aplicaciones. A título de ejemplo: alarma de intrusos, sismógrafo remoto o manubrio de robots contruidos con kits para

estudiantes. Estas son ideas frescas, más dignas de una clase de Tecnología, cuya concreción paso a paso puede encontrarse a "dos *clicks*" de *Google*<sup>32</sup>.

#### **3.1.4. En el país de los ciegos**

En una sociedad ávida de respuestas técnicas, cualquier tecnólogo avalado por un título suele rodearse de una suerte de aura moderna de mesianismo. El empresariado incluso ha bautizado mercantilmente como *consultores* a estos "mesías", debido a los milagros que les preadjudican.

Valga el siguiente ejemplo, ante la recurrente interpelación del alumnado respecto de los virus, los docentes de Tecnología suelen aportar como única solución la "adopción" o piratería de un antivirus. Es decir, normalmente recomiendan la instalación de un programa que hace la suerte de agente de seguridad, estorbando el 20% de las acciones de la CPU y que, además, está desfasado aproximadamente en un 60% respecto de las mutaciones posibles de los virus vigentes. Esta respuesta simple, que parece obvia, es inaceptable como única alternativa en una clase de Tecnología, ya que no resuelve el problema, sólo lo desplaza. Incluso se desaprovecha un momento maravilloso para estudiar un que se manifiesta como una explícita y consciente pérdida medida en energía y dinero. Esta pérdida, que paradójicamente es tan aceptada por el vulgo como lo es la aspirina para curar los resfríos, debería ser al menos revisada y discutida en clase<sup>33</sup>.

32 Anónimo (2014). "*Todos a la Robotica*". Recuperado de <http://www.todosalarobotica.ulp.edu.ar>, [consulta el 14 de octubre de 2014]

33 La razón de que los virus existan es que aprovechan vulnerabilidades en aquellos programas cerrados de los cuales no se dispone de su código fuente. Por el contrario, de disponerse de la fuente del programa, cualquier programador medianamente avezado, puede bajarlo del repositorio, emparchar la vulnerabilidad sin esperar a que lo haga la compañía, y luego publicar "la vacuna". En comparación, las compañías vendedoras de *software* reaccionan cuando la

Los docentes de Tecnología deben atacar el problema de fondo. *Los virus entran a los sistemas cerrados porque estos son binarios imposibles de arreglar.* Mostrar en su lugar los repositorios de paquetes de *software* que hay en *Linux*. Estos paquetes están chequeados vía MD5 todo el tiempo, impidiendo ulteriores sorpresas. Deben asimismo enseñar a desconfiar a los alumnos cuando se encuentran en *Windows* respecto de las respuestas de los buscadores. Es inadmisibile la falta de calidad de las respuestas de todos los buscadores, *Google* incluido, las cuales conducen a páginas con toda clase de artimañas para apelar a la ignorancia e ingenuidad del usuario. El alumno debe entender que un *click* en una firma para detener la matanza de focas bebe en Groenlandia, tiene por probable consecuencia un estado de permanente promiscuidad en su computadora, la cual quedará enlazada vía conexión inversa con otras miles, ejecutando como esclava ("*zombie*", en el *argot hacker*) para cumplir tareas de emisión de SPAM, alquiler de uso de CPU, o servicio HTTP de entrega de material pornográfico, por mencionar algunas de las tareas a las que Windows es sometido cada día de infección. Si el docente no es capaz de animarse a usar Linux, y no es capaz tampoco de descomponer una *url* hasta su *host* final, es decir no es capaz de distinguir la diferencia entre seguir un enlace que lleva a greenpeace.org, o del peligro de hacer *click* en la cara de una foca que conduce a (ejemplo) [http://200.44.22.13/deme\\_Su\\_Billetera.php](http://200.44.22.13/deme_Su_Billetera.php), pues al menos debe enseñar a los alumnos que tomen tres recaudos, a saber:

- desconfiar de toda dirección que no termine en ".edu" o ".org"; y

"infección" ya es masiva.

Una mejor respuesta, profiláctica, gratuita y socialmente inclusiva es: "Instale Linux, donde las infecciones en los escritorios es cercana a cero".

- recordar que para que una persona malvada del otro lado pueda dañarnos, necesita que nosotros primero intentemos hacer algo ilegal, o que al menos, colaboremos intentando obtener gratis algo que posee derechos de autor.
- recordar la técnica del vampiro; en las leyendas antiguas, los vampiros no pueden entrar a las casas. Por eso golpean nuestras ventanas ("*windows*") y nos seducen para que quitemos el pestillo. Actualmente el equivalente es ignorar las advertencias del sistema operativo, o de aceptar indiscriminadamente los frutos de la Internet.

En pocas palabras, los docentes deben inspirarse en los médicos que imparten Educación Sexual, quienes se focalizan en brindar medidas profilácticas ("previas"), en lugar de enumerar remedios "post-acto" o reparatorios de noxa (afección). O inspirarse en la técnica de los ciegos, que consiste simplemente en desconfiar del sentido del oído, tanteando antes de avanzar.

Otro ejemplo de ceguera, o cuando menos de falta de lucidez consiste en la recomendación del docente sin reservas de la adquisición de nuevo *hardware*. A esto se le llama "concentrarse en el medio", en lugar del objetivo. Si el usuario desea seguir creando textos o calculando sus cifras, debería poder seguir usando sus programas sin importar si su *hardware* se vuelve obsoleto. Debería poder contar con parches y versiones modernas de los programas, que corran sobre su computadora de siempre. Las empresas podrían cobrar un valor por estas actualizaciones, en lugar de dejar de mantenerlas; *Apple* es un buen ejemplo al respecto.

Sin embargo, la respuesta del tecnólogo mediocre sospechosamente es la misma que brindan tanto los neofílicos como los vendedores a comisión, la cual consiste en lo siguiente: "compre el *hardware* más caro que pueda pagar, y evitaré

nuevos incidentes", sin aclarar siquiera que la mayoría de los problemas del usuario se podrían resolver aumentando un poco la memoria RAM.

Lo mismo ocurre para las versiones de los programas. Si se interpela al docente tecnólogo, éste recomienda sin dudar la adquisición de las últimas versiones, no por ser más seguras y estables, sino porque son simplemente las últimas, aunque no lo haya constatado fehacientemente. Ni siquiera pregunta al alumno cuanto o que usa de su versión actual del programa, o en qué se encuentra limitados. Por regla general, los alumnos usan apenas el 20% de las funciones de su procesador de texto o de su hoja de cálculo, las cuales son perfectamente reemplazables por opciones libres como Libreoffice.

Para las empresas, la respuesta de los tecnólogos es la misma, aunque varía en cuanto a su magnitud. Como ejemplo, algunos administradores de sistemas, ante la dependencia de Wintel, muchas veces optan por adquirir un oneroso ESX VMWARE (un *Linux*) que simula máquinas reales, de tal manera que *Windows* corra sobre ellas. Nótese la gracia en esta paradoja. Puesto que los usuarios no quieren aprender a usar *Linux*, entonces se les brinda una simulación de *Windows*, ¡sobre *Linux*! Ello no está mal para una empresa en miras de crecimiento, pero definitivamente muy lejos de una visión potente, estable y eficiente. Se recomienda en tal sentido averiguar qué usan en los gigantes *Google*, *Facebook* o *Twitter* para sus sistemas supergigantes, sus *clouds* repartidos a lo largo de todo el planeta, quienes literalmente son lo más parecido a una conciencia global conectada que ha visto la civilización hasta el momento<sup>34</sup>.

34 Le ahorro la búsqueda: Linux para los balanceadores de carga, FreeBSD en los *clusters* de millones de máquinas, Linux y OS/X en las oficinas. Puesto que no necesitan antivirus, ahorran electricidad usando menos ciclos de CPU, y hasta pueden darse el lujo de modificar sus generadores para trabajar con energía solar.

El docente no efectúa una convalidación epistemológica de sus propuestas de enseñanza. En lugar de reflexionar sobre las fuentes verdaderas del problema, destaca sólo sus partes visibles. Hace lo mismo que un docente de Biología que no enseña las raíces de la planta "para no complicar al alumno". Puesto que toma sus soluciones de las torpes acciones del vulgo, todo lo aborda desde un lenguaje y manera *vulgar*. Esto impide el desarrollo de capacidades heurísticas y hermenéuticas de las situaciones. Hace que los alumnos confundan síntomas con enfermedades, pretende la mayoría de las veces emparchar lo existente, mediante la incorporación del último grito de la moda a costa del bolsillo de otros. La velocidad de estos automatismos algorítmicos negativos es tan vertiginosa que sólo haya su límite en la capacidad de respaldo económico del demandante del servicio. Dependiendo de su formación universitaria, el tecnólogo no suele implementar una revisión formal del problema, puesto que generalmente el problema es él mismo.

Teniendo por defensor a Eduardo Escalante Gómez (2002), quien señala que en las últimas décadas han situado las investigaciones sobre la metáfora en el centro del pensamiento científico (*Mentes nómades*, 2002, pp. 126), es que voy a emplear la metáfora de Narciso.

Centrémonos en el impreciso *momentum* cuántico en que el informático es un usuario *con un título que afirma que es informático*. Esta legalidad le aporta la autoridad suficiente para instalarse como "Narciso en la torre de marfil". Y esto es un verdadero problema, puesto que a pesar de que recomienda mal, sus soluciones relativamente funcionan. Esto se conoce como "llegar a la conclusión correcta partiendo de una premisa falsa". En lugar de meditar sobre el camino recorrido, de refactorizar —optimizar— el método, se adjudica los pseudoaciertos y se ve al espejo como *informático "consultor"*, e incluso "referente", puesto que incluso aconsejando mal, acierta mediante la fórmula simplista de aportar acciones

radicales e innecesarias para problemas nímicos o baladíes. Con el tiempo, estas soluciones se convierten en la sempiterna respuesta a cualquier problema, cuando en realidad han cumplido al objetivo a la manera en que se llega a una conclusión correcta mediante un razonamiento equivocado.

Ello genera representaciones sociales negativas que conducen a los usuarios a desconfiar de cualquier informático, puesto que ellos mismos podrían haber arribado a la misma conclusión: "cambiamos el equipo", "reinstalemos todo de nuevo", "¿hiciste un *backup*?", "hay que formatear el equipo", así como otra serie de muletillas propios de quienes se presentan como expertos, pero que carecen de verdadera pericia. El usuario, como norma, debería aprender a desconfiar de los informáticos que solo formatean sin buscar la razón cabal del problema.

### **3.1.5. El *failsafe* del incapaz**

¿Acaso se puede resolver un problema sin abordarlo ni entenderlo? Por supuesto que sí. Actualmente las personas lo hacen de manera frecuente, al contratar expertos con el dinero que han ganado mediante el uso de otras aptitudes, que les permiten resolver problemas de otras personas. Así, los abogados contratan a plomeros, y los plomeros contratan abogados.

Sin embargo, en este punto del documento, el objeto de estudio se traslada al problema ético que acarrea: un dilema que debe afrontar el poseedor de la información, puesto que quien lo escucha (el alumno que más tarde aplicará las mismas soluciones), procede basado en recomendaciones erróneas y es, por lo tanto, inimputable.

Inmerso en su trabajo de capataz de línea de montaje buscar / copiar / formatear / imprimir, el docente de tecnología tiende a elevar a la categoría de

ultraje todo atasco, a indignidad todo acto de descenso al origen de cualquier problema técnico. Si la pregunta es planteada por un alumno, y no conoce la respuesta, este adquiere inmediatamente la condición de *enfant terrible*, una subversiva piedra entre los engranajes, un freno a la producción.

Al docente ortodoxo de tecnología no le gusta asumir la tarea de buscar la respuesta, ni de diferir la resolución fuera de hora. No sospecha de sí mismo cuando su contestación no posee un marco mayor que el aportado por un reparador de PC, cuya escueta respuesta suele ser: "Hay que formatear el disco". Y esto a riesgo de ser considerado como una persona que no ofrece respuestas fáciles, lo cual no debería ser una sorpresa, pues no hay respuestas fáciles. Cualquier técnico en PC puede corroborar que no hay respuestas fáciles cuando debe dar explicaciones a un superior en lugar de dársela a un usuario.

No hay una reflexión consciente de este acto. Solo hay un rosario de autojustificaciones, mediadas por el sueldo o el exceso de horas, tal que permiten procrastinar indefinidamente el acto de aportar *calidad total* a los conocimientos impartidos en clase.

La excusas son infinitas, al punto que se convierten en una suerte de refugio, al estilo del *failsafe* al que recurre frecuentemente Bruner (2004), y que describe como un dispositivo que opera como un sistema capaz de ajustar automáticamente una falla o error, mediante la detonación de un aparato autoprotectivo. Es decir; la puesta en juego de un mecanismo de autojustificación basado en un conjunto más o menos rudimentario de excusas automatizadas, tales que evitan analizar en mayor profundidad las situaciones, esquivando el objetivo de "resolver el problema" mediante al acto de degradarlo a "resolver el problema que supone encargarse de resolver el problema".

El docente busca quedar a salvo en su "integridad" de experto, y mantener su autorepresentación social de "resolvedor" de problemas asociados al mundo tecnológico. Por otro lado, busca ser reconocido y reconocerse a sí mismo como diestro en cuanto a la resolución efectiva de problemas informáticos. Sin embargo, resolver un problema no significa necesariamente haber descubierto la causa que lo desencadenó. Si se trabaja en el plano de los fenómenos consecuenciales, nada garantiza que el sujeto infiera las causas o motivos subyacentes, pues para ello es menester que se apliquen una serie de procedimientos de convalidación que, como habíamos mencionado, clarifiquen la heurística y correspondiente hermenéutica del caso o problema acaecido. Esta convalidación requiere de diversos expertos de su propio tiempo. Sin embargo, en las entrevistas realizadas trascendió que pide ayuda solamente a amigos y conocidos, figuras de autoridad, del tipo que aparecen en las introducciones de epistemología como fuentes cuestionables de conocimiento, en lugar de expertos que pueblan foros y canales de IRC. Consecuentemente, tampoco ayudan en foros ni se mezclan en proyectos comunes de construcción de software.

### **3.1.6. El bloqueo**

De acuerdo a las observaciones de campo, mientras los docentes "no tecnológicos" *preguntan* cuando se encuentran con algún tipo de situación sin poder resolverla, los "tecnólogos" en cambio asumen posición de *alone wolf* y se bloquean. A menudo carecen de base para distinguir cuándo están solo consumiendo algoritmos simplificantes y reduccionistas que obturan el pensamiento lateral en el sentido empleado por Edward De Bono, siguiendo procesos "seguros" y rutinarios "salvadores" o "confiables". Estos etiquetados en algunos casos funcionan, pero en otros, dificultan notablemente el abordaje de situaciones que "escapan a la regla" o matriz de acción estandarizada. La

perspectiva marcuseana del hombre unidimensional opera aquí como pensamiento simplificador, reduccionista, matricial, holístico (aunque en un sentido malentendido) y algorítmicamente deficitario (por emplear algoritmos que no incluyen rutas que incorporen emergentes, riesgos e incertidumbres dentro del sistema).

Uno de los principales efectos es el hecho de eliminar las anomalías como aspectos despreciables, en vez de buscar hipótesis *ad hoc* o *ceteris paribus* que ofrezcan alternativas, que en caso de funcionar, puedan salvar el sistema desde una perspectiva racional, o bien animarse a reemplazar el algoritmo por otro que permita dar sentido al nuevo sistema que se configura de manera emergente al observador atento. No se trata por lo tanto de desplazar el problema para el futuro: esto provoca una estructura de desplazamientos o postergaciones del conflicto para un futuro del cual se huye y que se pospone como un horizonte al cual aparentemente nunca se llega, muy conocido por los psicólogos como un mecanismo neurótico. Antes bien, cabe plantearse la detección de bucles de nivel 1 según Watzlawick, tal que no se limite simplemente a resolver el problema de la reducción, sino también el del desplazamiento.

Es menester que el docente examine metacognitivamente sus propias prácticas. No obstante este examen depende del grado de conocimiento teórico y práctico que el mismo posea y de su desarrollo competente. Tras una autoevaluación, ésta desencadena un proceso de realimentación. Esta puede ser positiva o "de refuerzo" en la medida en que el sistema alcanza sus propias expectativas o misión telética, pero también cabe que el sistema se haya degenerado, cuando ha dejado de ser beneficioso para su cometido. En este caso la realimentación exige un proceso de ajuste y transformación que requiere una identificación de los "puntos de palanca" del sistema considerado según McDermott y O'Connor.

En sistemas, un bucle con realimentación positiva permite mantener la homeostasis del sistema; en cambio, un bucle con realimentación negativa pone de manifiesto la negentropía del sistema y la posibilidad de una reorganización y autosuperación si se resuelve con un bucle que pueda recapitalizar las falencias o déficits y generar un algoritmo innovador y adaptativo que resulte superador respecto de las anomalías emergentes y que de lugar a una compensación. Siempre atentos a incluir mecanismos de corte cuando construyen algoritmos en bucle, los informáticos deberían ser los primeros en desconfiar de la insistencia en patrones de aprendizaje. No siempre el adagio "persevera y triunfarás" es verdadero, especialmente si la única fuente de procedencia del conocimiento es el mero hecho que todos los demás docentes solucionen los mismos problemas de la misma manera, un sesgo del conocimiento llamado *argumentum ad populum* pero empleado por el cerebro para autoconvencerse, en función de conservar energía.

La carencia de pensamiento sistémico es tan patente como la carencia de formación epistemológica en la carrera de Análisis en Sistemas. En Epistemología, por ejemplo, se aprende a desconfiar de las respuestas establecidas, incluso si éstas cuentan con gran aceptación de la comunidad científica: incluso si todo el grupo la considera válida, todos podrían estar equivocados, si se lo analiza desde un nuevo paradigma o programa de investigación.

Por lo tanto, el pensamiento de los profesionales en sistemas, paradójicamente no es sistémico, sino algorítmico, propios del hombre-cyborg de Burbules. Contemplar un robot cavilando sobre sus propias limitaciones, y verlo "colgado" en el proceso, es natural para un modelo de inteligencia artificial atado a algoritmos restringidos y preinsertados o programados. Distinto es concebir un robot capaz de innovar (transformar sus algoritmos para resolver problemas de formas inusitadas), coincidente con la idea de salto hacia un nivel 2 de Watzlawick (el cerebro positrónico de Isaac Asimov planteado en la película de ciencia ficción

*Yo robot*). Pero contemplar un informático atrapado en los mismos bucles de resolución es cuando menos perturbador. Esta rigidez es producto de una matriz cultural de resistencia al cambio, y potencialmente patológico. Por ello es recomendable cultivar una serie de actitudes que abran posibilidades a los docentes en el campo de las NTIC, a saber:

- *pasión*: bucle de realimentación por excelencia como oposición al desinterés, el cual pasa a ser uno de los principales obstáculos para la autosuperación de los enfoques restringidos. Se trata de descubrir lo apasionante y disfrutar de ello como un desafío lúdico recreativo, tal que reporte beneficios casi por accidente. Es decir, no poner los beneficios económicos como razón primera. Para ello es menester autoconvencerse y automotivarse. Las NTIC son un campo vasto y nuevo que reclama espíritus ávidos de saber, sujetos que se sientan cómodos con búsquedas heurísticas. Para respuestas puntuales, precisas y falibles están los buscadores (máquinas) y los predicadores. En el análisis de datos se trabajará este tema como CM (Componente Motivacional);
- *desconfianza*: el bucle de compensación aplica la duda metódica cartesiana (desconfiar metódicamente de lo seguro y preestablecido), por lo cual si todos proceden de cierto modo es probable que ese camino no sea el verdadero o el superador de la situación presente. Sospechar todo el tiempo si se forma parte de la nueva ola, o si se forma parte de una banda de lemmings desesperados. Generalmente, los problemas se trascienden reenfocando los mismos desde una nueva perspectiva, incorporando nuevas variables o redefiniéndolas, procesándolas por medios alternativos y, en general, ampliando los horizontes de búsqueda;
- *sincretización*: entendida como la incorporación de elementos diversos y heteroclíticos, ajenos a la generalidad imperante (así como el hierro mejora

cuando se le introducen impurezas). Muchas ciencias han dado saltos teóricos cuando han introducido conceptos surgidos en los campos de otras. La lateralidad del pensamiento tan difundida por De Bono es una fuente prometedora y auspiciosa de buenas ideas o actitudes favorables. Así como el médico recomienda el ejercicio para mejorar cualquier actividad mental, la inclusión de ciertas artes como el teatro o de ciertas actividades como el voluntariado y la solidaridad, pueden ayudar a cambiar estos estados circulares<sup>35</sup>;

- introducción de ruido: contraviniendo los esmerados procedimientos de retroalimentación positiva, que forman parte de los supuestos que utilizan los docentes como formas de protección ante el caos, como una forma consciente de mejorar su propia adaptación. La manera más simple es saliendo de la zona de confort. Watzlawick llama a esto "la práctica del cambio" (1992, pp. 135);
- aplicación de un racionalismo crítico: se trata de seguir la propuesta kantiana de enjuiciar a la propia razón en su proceso y condiciones de posibilidad (escrutinio de sus propias capacidades) y volver a las teorías que tratan la complejidad, bases para cualquier estudio de sistemas. Para la carrera de sistemas se considera solamente aquella inteligencia lógico-matemática y lingüística que Gardner (2004, pp. 45) describe (y defenestra) como "inteligencia basada en la curva normal" —visión clásica según la cual el CI (cociente intelectual) determina la inteligencia del individuo y su éxito en la vida—. Es fácil de comprobar cuando se detecta que la manera de "elevar la calidad del egresado" en las carreras de sistemas es meramente ajustando las horas de matemática. Lo cual no es del todo incorrecto: un

35 En realidad la solidaridad genera la liberación de oxitocina, útil para ciertos mecanismos asociados a la resiliencia (McGonigal, 2013).

alto nivel de CI aumenta las posibilidades de una rápida carrera, y hay que admitir que los egresados de ingeniería son muy buenos proyectistas y calculadores, pero esto no asegura la capacidad de crear variables "imaginarias", de establecer conexiones laterales, armar conjuntos temporales de supuestos que completen las partes de otras ecuaciones en otros desafíos técnico-tecnológicos (innovar y crear).

Entonces, ¿en dónde se puede hallar el componente actitudinal faltante? Antes de responder a esta cuestión, cabe examinar en líneas generales la propuesta gardneriana de las inteligencias múltiples:

- inteligencia verbal-lingüística (entendida como la habilidad de usar palabras y el lenguaje);
- inteligencia lógica-matemática (definida como la capacidad de pensamiento y razonamiento inductivo-deductivo, al igual que de cálculo, reconocimiento y abstracción de patrones numéricos o formales);
- inteligencia espacial (considerada como la capacidad de ubicarse espacialmente según sistemas de referencia autoreferenciales o no, así como de percepción de las dimensiones de cuerpos, creación de imágenes y cuadros internos organizadores de objetos en el espacio);
- inteligencia cinética (definida como la capacidad de dominio del propio esquema corporal y sus posibilidades dinámicas)
- inteligencia musical (consistente en la capacidad de reconocer patrones de tonos, sonidos, ritmos y traducirlos en distintos dispositivos sonoros, o bien la capacidad de ejecutar a voluntad la traducción instrumental de una vivencia sonora intrapsíquica o captada del ambiente circundante);

- inteligencia interpersonal (concebida como la capacidad para establecer y resolver conflictos sociales, y establecer comunicación persona a persona de manera altamente eficiente y de calidad); y
- inteligencia intrapersonal (asociada a vivencias espirituales inherente al propio comportamiento, la autorreflexión, la conciencia, y la autorrealización de los sujetos).

Cabe plantear una aplicación de la teoría de las inteligencias múltiples gardneriana para la dilucidación de una serie de aspectos, a saber:

- a) entender que no se puede resolver el problema entrando en bucle de realimentación positiva, puesto que los docentes de tecnología vuelven a generar el mismo error varias veces hasta darse por vencidos, creyendo que si insisten —del mismo modo que hacen los usuarios legos—, el problema se resolverá por sí solo. Este comportamiento se parece al juego del ratón narrado en la novela de Orson Scott Card *El Juego de Ender*, donde, en una metáfora, el protagonista debe realizar a la fuerza acciones de pensamiento lateral para sortear su bloqueo;
- b) la inteligencia intrapersonal propicia instancias de metacognición y autorreflexión para entender qué se está haciendo y superar procesos de pensamiento lineal que conducen a una aporía, por seguir interfases autoconducentes que llevan a un pensamiento vertical (De Bono, 2005). Para sortear esta suerte de obstáculo epistemológico en sentido bachelardiano (Bachelard, 2000) es preciso establecer un acto creativo: "El pensamiento vertical solo se mueve si hay una dirección en la que moverse; el pensamiento lateral se mueve para crear una dirección" (2005, pp. 48). Por tal motivo tener una visión global y no individual del problema es un paso para lateralizar o advertir alternativas producto de una capacidad

reflexiva sobre los propios procesos cognitivos (metacognitivos). Si todo problema es una cuestión de magnitudes, entonces es poco probable que alguien sea la única persona bloqueada ante el mismo problema. En alguna parte de la Internet debe estar tratado el tema y esos extraños códigos tabulados<sup>36</sup>; y

- c) quien presenta un notable desarrollo de la inteligencia lógico-matemática, suele presentar un menor desarrollo de las inteligencias Interpersonal e Intrapersonal, por la simple razón que la capacidad para abarcar objetos abstractos de estudio sin mayores dificultades, involucra menos necesidad de consultar opiniones. De allí a que tienda a consultar o a participar menos de espacios de socialización en redes. El desarrollo de estas inteligencias gardnerianas, es necesariamente complementario para desarrollar la gimnasia de pedir ayuda<sup>37</sup>.

Resulta a veces difícil de comprender que el docente de Tecnología o de Informática no haya considerado, aunque más no sea por un momento, lo que hacen algunos alumnos ante este caso: recurrir al encargado de laboratorio o a otros docentes de Tecnología. Sin embargo, algo traba el proceso de transferencia horizontal de solicitud de ayuda, y el docente prácticamente entra en pánico, análogo al del conductor que se pierde en una zona nueva, y que se niega a preguntar a gente de la zona.

¿Por qué los docentes de informática no piden ayuda al encargado de laboratorio o se limitan a culpar a otros del problema? Si se trata de un problema técnico, ¿acaso no sería ideal que emerja precisamente en una clase de

36 Para el ejemplo de las pantallas azules, por más atemorizantes que se vean, todos sus códigos hexadecimales de error están descritos en los foros de Microsoft.

37 Luego se volverá sobre el tema de las formas y los protocolos de los pedidos de ayuda.

Tecnología, donde se lo pueda tomar como tópico generativo de situaciones de aprendizaje múltiples y altamente significativas?

Watzlawick no duda en situar el tema como un problema de ambivalencia en la comunicación (1992, pp. 51-9). McDermott y O'Connor lo distinguen como Proalimentación (1998, pp. 73) o "efecto anticipado del futuro" (rechazo, falla, vergüenza), que todavía no ha tenido lugar, y que genera la causa del presente. Entre las observaciones en clase efectuadas como Asistente de Trabajos Prácticos, se destacan a) miedo a otros especialistas del área (miedo a no ser suficientemente competente ante los pares); y b) miedo a que los alumnos descubran sus falencias (complejo de inferioridad puesto en evidencia ante la falta). Naturalmente, a la manera del gato de Schrodinger, la sola presencia del observador, por más invisible que intentara ser dentro del aula, contamina la muestra.

No obstante la observación es válida desde el momento que el docente de informática y tecnología trata de preguntar lo menos posible al ATP. Y cuando se ve forzado a hacerlo, lo hace a regañadientes. Normalmente, la persona *que no pregunta* lo hace para no evidenciar su falta de competencia en el tema, y es común que los profesionales no consulten a otros profesionales para no evidenciar su falta de dominio sobre el tema en cuestión.

En informática, dentro del circuito comercial, el problema de la comunicación entre profesionales e incluso entre miembros del personal es una patología endémica y bien conocida. En empresas convencionales se utilizan diversos métodos para comunicar las áreas, desde proponer reuniones informales a inscribir a todos en programas de coaching ontológico. En las empresas de informática se utilizan metodologías ágiles, siendo *Scrum*, la más conocida, para promover la participación de todos los miembros del equipo. El objetivo es el mismo: evitar serios atascos en los procesos.

Por lo tanto no preguntar es típico de ambientes competitivos propios de empresas ortodoxas, con la que se traslada al aula de Informática, donde los docentes, o bien no son docentes a tiempo completo (que trabajan en empresas), o bien tienden a acarrearse ese mismo tipo de comportamiento, debido probablemente a representación social fuertemente acendrada que concibe al informático como alguien que debe poder resolver todo tipo de problema propio del campo, e incluso, de campos afines. A excepción de los profesores en informática (con título específico), muchos docentes tienen a la docencia solo como alternativa económica, mientras buscan trabajo en otras empresas. De esta manera no sólo *no preguntan*, sino que llegan a defender tenazmente sus argumentos, no por ser estos válidos, sino, por todo lo contrario, para que no se descubra la ignorancia de los mismos. Persistir en un argumento a pesar de tener todas las demostraciones en contra, es precisamente, el comportamiento de los necios.

El gran error que los docentes de tecnología incurren en este caso, es ignorar explícitamente que las computadoras están llenas de errores, y que todo error emergente puede ser capitalizado didácticamente en una situación de aprendizaje significativo de alta pregnancia cognitiva. Un cuadro de diálogo con un símbolo de exclamación (o "error") que aparece en la pantalla de una terminal es muy valioso para llamar la atención al resto de la clase, e inferir entre todos qué es lo que podría estar precisando la computadora para continuar su proceso; al fin y al cabo a cualquiera de los alumnos podría haberle ocurrido.

Establecer, convalidar y asimilar socio-comunitariamente que nadie sabe todo, es el primer paso para una construcción sinérgica del conocimiento y un principio de una salud psicosocial proactiva. El gran error es dar una clase con alumnos sincronizados (todos realizando la misma tarea y pretendiendo avanzar simultáneamente en el mismo paso, como una cadena de producción al mejor estilo fordiano), donde si un paso no se logra, entonces se "atasca" o detiene todo

el proceso grupal de aprendizaje (cadena de producción). Como a principios de la revolución industrial, entonces habrá que buscar al "maquinista" (el encargado de laboratorio), porque uno de los operarios (o el capataz) se ha caído dentro de los engranajes.

¿Cómo se puede resolver el problema de pedir ayuda? Quizás la solución se encuentre dentro de la misma aula. Una técnica que puede implementarse con alumnos competentes pero tímidos, es mejorar sus habilidades sociales induciéndolos a que ayuden a sus propios compañeros. Esto redundará en una rápida y visible mejora de comportamiento del alumno y en un incremento en sus habilidades de comunicación al sentirse útil. Este truco para mejorar el desarrollo del grupo no sorprenderá a profesores (con título de profesores) que han estudiado la Zona de Desarrollo Próximo de Vygotski, pero puede que contribuya e incluso sorprenda a docentes de informática que tienen a los alumnos competentes como incómodos *enfant terribles*.

### **3.1.7. El necio encubierto**

La ineptitud de los *alone wolves* suele pasar desapercibida a los directivos y pedagogos de las escuelas, donde no hay examen de competencia y no se exige mayor credencial que las que otorga un bono de puntaje. De alguna manera se presupone que estos docentes lacónicos son talentosos, y que mantendrían una supuesta reserva de sus saberes por la complejidad que revisten los mismos; si contestan en forma limitada, es porque contienen su caudal de conocimientos para no ahogar al interlocutor. Se presenta como el docente paternal: da "lo justo y necesario".

La puesta en evidencia o detección de la falta —según la categoría psicoanalítica— no se patentiza tan fácilmente. Si los directivos se ven

imposibilitados para medir sus habilidades, los asesores pedagógicos no poseen competencias técnicas para detectarlos, los padres no pueden tomar contacto con las clases en sí y los colegas docentes no suelen comprometerse con denuncias de compañeros, entonces sólo queda que los alumnos puedan ponerlos en evidencia.

Sin embargo, y a pesar de lo que teme el profesor novel, raramente los alumnos descubrirán sus carencias. Ello no se debe a que carezcan de ímpetu al respecto, sino porque raramente poseen las habilidades que se les atribuyen. Jakob Nielsen (2013), respecto de los jóvenes y del esfuerzo puesto en construirles *interfases* especialmente fáciles de usar, sostiene:

"Los adolescentes no son tecno-magos que surfean con displicencia la web [...] Los adolescentes a menudo son estereotipados como personas que buscan solo cosas que sean llamativas y diferentes. A menudo son considerados como arrojados ante la tecnología, y constantemente conectados a algún medio. A pesar que estas aseveraciones son parcialmente verdaderas, también son demasiado simplificadoras. Adecuar sus diseños a ellas puede llevar a desastrosos resultados."<sup>38</sup>

Me permito ir incluso más allá de Nielsen, pues los estudiantes son aún más tontos usando tecnología que los mismos adultos. A veces surge el comentario respecto de que las nuevas generaciones que *ya vienen sabiendo, he incluso reciben la denominación de* nativos digitales. Sin embargo, es dudoso que puedan formar parte de un salto evolutivo (*gap* en su ADN) sucedido en una sola generación, es decir, una mutación científicamente no constatable al modo de los niños "índigo". Basta con observar la cantidad de virus y troyanos que tiene una

38 El párrafo original se ha traducido arbitrariamente como "*Teens are not technowizards who surf the web with abandon [...] Teens are often stereotyped as only wanting things that are bold and different. They're also often viewed as being fearless about technology and constantly connected to some form of media. Although this might be partially true, it's an oversimplification and letting this steer your design can lead to disastrous outcomes*".

computadora de un estudiante para evidenciar su incompetencia en un uso inteligente y hábil de las NTIC. En comparación, los adultos al menos tienen desarrollada (o deberían tener) una mínima desconfianza: intuyen que lo escrito en una página *web* posee el mismo valor que tiene un papel lanzado por debajo de una puerta. Pero sus hijos son increíblemente confiados en tal sentido. Si la hoja de papel ordena al adolescente que la vuelva a lanzar por debajo de la puerta, acompañada de fotos íntimas, aceptación de leoninos términos de uso, e incluso, datos filiatorios, fecha en que se encuentra de vacaciones u horarios en que puede ser interceptado por la calle, éste lo efectuará sin mayores reservas. E incluso se sentirá feliz por la posibilidad de darse a trascender, sonriendo cuando un *widget ajax* termina de subir sus fotos personales, feliz como una foca amaestrada que le sueltan una pelota para que la balancee sobre su nariz.

En conclusión, puesto que, en general, el estudiante no está en condiciones de detectar las carencias de los docentes de tecnología, estos pasan muchas veces desapercibidos dentro del sistema, gozando además de una representación social positiva. Estas representaciones permiten que muchos gocen de privilegios que no lograrían en otros espacios de conocimiento.

### **3.1.8. El problema de la escuela vista como una empresa**

Potencialmente el docente de informática se encuentra frente al aula, no para cumplir su deseo de transferencia de conocimientos, porque simplemente no hay lugar para él en otros ámbitos. Esto no debería ser estigmatizado. Se aprende también enseñando, y se puede crecer profesionalmente tanto en una escuela como en una empresa, pero para lograrlo, se debe entender que los espacios son distintos. En educación, el intercambio de conocimientos es todo. El docente de informática que así lo entiende es fácil de detectar, ya que es una persona

proactiva, que se lleva bien con estudiantes y profesores, propone talleres y actividades, y fundamentalmente disfruta de su trabajo.

En cambio, el docente de informática que no ha encontrado un modo de vida en el campo profesional empresarial, y lo busca en la escuela, es fácil también de percibir porque se mantiene a la defensiva. Trata a la escuela como una empresa y a los estudiantes como usuarios. Tampoco enseña: sólo entrena. Vislumbra el conocimiento como un bien a resguardar y sus "lagunas" sobre el mismo como vulnerabilidades que hay que ocultar frente a sus colegas. Marca tarjeta y corre a otra escuela.

En ocasiones los docentes de tecnología trabajan también en empresas. Esto es bueno porque del lado empresarial puede nutrirse más rápido respecto de innovaciones y de las necesidades del mercado. Pero por el otro se aprecia un punto ambiguo en lo que se refiere a la conciliación de su vida técnico-tecnológica profesional con la pedagógico-didáctica profesional. Esto se detecta en que no suelen exponer sus carencias cognitivas dentro de la escuela, para que sean resueltas mediante el intercambio con colegas del mismo campo, de otros campos o de los mismos estudiantes. Del lado de la empresa ocurre otro tanto: suelen ocultar su desempeño en el ámbito pedagógico-didáctico como si se tratara de una "caída" a nivel profesional, que solo legitima o valorada si se trata de docencia en contextos de educación superior.

Lo que en muchos casos resulta recomendable en el plano empresarial (reserva de información) en el ámbito educativo es considerado como una falta ética al ejercicio de la profesión. Esconder el conocimiento entre empresas es natural y comprensible. Pero hacerlo dentro del ámbito educativo es paradójico e inclusive contraproducente, pues un docente es mejor docente en la medida en que sus estudiantes aprenden más y mejor. Si el docente ha llegado a la escuela

debido a falta de experiencia o conocimientos, debe entonces entender que ha llegado más *para aprender que para enseñar*.

Dar la espalda al otro, es darle la espalda a la diversidad, al interjuego transformador de dos conciencias que se conectan y dan lugar a algo que las trasciende. Cerrarse al otro es obturar la mirada, es permitir una esclerosis de la misma forma que se rehúsa a considerar otra cosa que no sea aquello que puede ver y que tiene ante los ojos, sin estar dispuesto a percibir otros escorzos que reperspectivicen la mirada. Estas reperspectivizaciones debieran dar lugar a una flexibilidad del pensar que se abre a lo alternativo, a "y si se viera así, qué sucedería". La generación de experimentos cruciales es un canto a la imaginación frente a un universo infinito de lo posible, que luego se conjuga con lo deseable y se termina afinando en lo factible. Pero sin este despliegue, difícilmente se puede hablar de creación, de producción y de innovación. Entonces, siguiendo la presentación presentada por Ferriz Olivares (1985), ¿cómo enseñar a pensar la tecnología sin desarrollar un pensamiento abierto a la complejidad de lo pensable, la consideración de escenarios futuros (propios de la prospectiva), y el sentido inter y transdisciplinario que mueva a la colectivización de la búsqueda?

## **3.2. Bloqueo entre los bloqueos**

### **3.2.1. Introducción**

Ya se había tratado el tema del bloqueo<sup>39</sup> anteriormente y de la importancia de *aprender a dejarse ayudar*. También señaló la importancia del pensamiento lateral y de crear el camino cuando este no existe<sup>40</sup>. Por tal razón, en este capítulo, se ahondará en dos tópicos, a saber: por un lado existe el problema de no manejar

39 Punto 3.1.6 *supra*.

correctamente protocolos solidarios de ayuda y, por otro, el hecho de que la situación de bloqueo se suele dar en contextos de mayor escala, que se tratará a continuación.

Hay situaciones en la vida que provocan una suerte de bucle del cual es difícil salir. Una de estas situaciones se da cuando el profesional en informática que no encuentra cabida en las empresas, elige la docencia como refugio económico. En este punto, si persiste en tratar al aula como una línea de montaje, a los alumnos como usuarios y a los demás docentes como competidores, la condición *temporal* de docente bien puede convertirse en *permanente*. El problema empieza cuando consideran que el aula es un espacio limitado, cuando en realidad el límite es personal.

El espacio áulico puede ser un excelente lugar para desarrollar incipientes habilidades, varias de las cuales inusuales en el perfil "tímido" de los informáticos tradicionales: oratoria, inteligencia emocional, colaboración, actitud de servicio y, especialmente, para desarrollar la costumbre de investigar. En las aulas se pueden plantear talleres y proponer múltiples actividades. Los alumnos, sobre todo *les enfants terribles*, si bien pueden poner a prueba el ingenio y las ganas de investigar del docente, también pueden convertirse en excelentes aliados. Junto a los alumnos, se puede aprender de todo: servicio técnico, soporte, programación de pequeños sistemas que ayuden a la planta funcional, instalación de redes, instalación de servidores simples, automatización de tareas rutinarias, entre otros. Todo puede realizarse con muy poco impacto negativo, a diferencia de las empresas donde si sale algo "mal", entonces ello se traduce en pérdidas económicas por las cuales puede ser motivo de despido.

40 Ejemplo: si la pregunta no existe en ningún foro, plantearla por vez primera, en lugar de abandonar el intento. Linux es un buen ejemplo de sistema operativo reactivo a tecnólogos con baja tolerancia al fracaso, ya que se abandona a los pocos intentos de no encontrar caminos conocidos. En el siguiente capítulo se trabajará este tema.

Si se llega al punto en que se ha acumulado suficiente experiencia como docente de Informática, entonces es muy probable que ocurra "el llamado de las empresas" por sí solo; es la madurez propia de una construcción paulatina de la experticia producto de una serie de factores asociados al compromiso, la pasión por la tarea, la perseverancia en la resolución de bloqueos y la creatividad constante. La fama de un buen docente de informática trasciende las fronteras de las escuelas. Tarde o temprano ocurrirá la "entrevista soñada", en la cual el docente podrá demostrar su competencia en varias herramientas, producto del residuo cognitivo obtenido de las clases y el conjunto de conflictos cognitivos experimentados y capitalizados como saberes profesionales dentro del campo.

En ocasiones el docente de informática que no quiere ser "docente" mantiene en el aula la misma actitud que lo trajo a la escuela. Se sube a su torre de marfil, marca tarjeta, no propone, no ayuda, ni aprovecha la obra social, ni el sueldo, ni aprovecha los espacios para aprender "sin consecuencias", se mantiene limpio de todo residuo cognitivo y, en términos generales, se comporta como un falso *alone wolf*<sup>41</sup>. Si a estos comportamientos se les suma el hecho de que no sabe pedir ayuda cuando se bloquea, entonces su condición puede llegar a ser la misma de Robinson Crusoe, es decir, un náufrago moderno varado en un aula.

### **3.2.2. El Nudo Gordiano**

El docente de informática (en rigor, cualquier ser humano) no se siente cómodo exponiendo sus lagunas de conocimiento frente a los alumnos o frente a otros profesionales. Sentimiento que requiere de introspección para admitir que se corrige simplemente a) siendo honesto intra e inter personalmente; b)

41 Recuérdese que la concepción de *alone wolf* empleada aquí, descrita en su propio capítulo, *refiere a* una persona que tiene competencias logradas en su campo.

preguntando; y c) investigando las respuestas. Por lo tanto, el docente de informática en peligro de atasco es fácil de detectar, porque no ha realizado una epistemología de las condiciones de producción de los datos en los que se basa. Esto ocurre porque a pesar de contar con una gigantesca esfera de datos<sup>42</sup> adonde recurrir, y como se puede observar en las entrevistas analizadas más adelante, tiende en cambio a consultar *solamente a personas con las cuales tenga un alto grado de confianza*.

Esto supone un problema, porque ciertas situaciones requieren soluciones muy de fondo, que no se encuentran en la consulta popular de los informáticos amigos. Y especialmente no se encuentran en los técnicos o en personal de soporte, quienes tienden a aportar soluciones que no tienen relación con el alcance pedagógico que requiere la solución. Por ejemplo ante un funcionamiento anómalo en un equipo cuyos síntomas son lentitud, barras de herramientas y mensajes conducentes a acciones de pago, los técnicos tienden a esquivar la causa del problema, y directamente hacen *backup de todo* el equipo (o no, para desgracia de los usuarios) y lo *formatean*. Tal cual hace y recomienda el docente de informática.

Con este conocimiento, el profesor de tecnología se presenta ante los alumnos como otrora Alejandro el Magno, quien desafiado a desanudar el nudo gordiano, enarbola la espada y lo corta de un tajo.

Estos problemas técnicos son un buen ejemplo de situación que deben resolverse de fondo, so pena de volvérselo a encontrar. Esta resolución debe realizarse buscando opiniones más expertas, o teniendo la valentía de ser el primero en preguntar si no se encuentra un caso similar. Sin embargo, y he aquí una nueva paradoja. Si bien el ambiente aúlico es rico en preguntas y respuestas, el docente no pregunta donde pueda exponerse. Incluso escondido bajo el

42 Término acuñado por Dan Simmons para su saga "Los Cantos de Hyperion".

anonimato de un "alias", raramente el docente de tecnología se acerca a preguntar a foros o listas de correo especializadas. Además, en un punto, si no se siente cómodo en estos espacios de consulta, hasta se puede dar un efecto *boomerang*. Este efecto puede medirse por la nula participación que hace el docente de tecnología respecto de los cientos de comunidades que están involucradas directa o indirectamente con la educación, la tecnología, el *software*, y especialmente, el *software* libre, el cual es por necesidad, rico en transferencias entre pares.

En las universidades argentinas, como ejemplo de lo expuesto, no se induce a acceder vía protocolo *IRC* a redes como *Freenode* (conocidas en el ambiente científico) y a participar en sus ricos canales de charla. En Europa este tema no solo se enseña, sino que son las mismas universidades quienes solventan toda la infraestructura de estos canales informales.

### **3.2.3. Pensamiento lateral creativo: el problema de crear la solución**

Muchas veces la respuesta no está en los buscadores. *Google*, a pesar de sus algoritmos de búsqueda, no es todo lo "sabio" que debería ser. Esto se encuentra en conocimiento de sus mismos creadores, ya que trabajan para mejorarlo desde una perspectiva semántica que nos lleve a una atrasada *Web 3.0*. *Google*, por ejemplo, no registra conversaciones en la red *IRC*, ni propone sitios limpios de virus desde donde bajar *software*. Guiado por normas *SEO*, o por razones propias, por ejemplo, pone al peligroso *Softonic* como sitio donde bajar antivirus.

De hecho, si *Google* fuera tan "sabio", compraría derechos de indexación (y los mismos textos) de revistas científicas, en lugar de traer comentarios obtenidos en *Yahoo* Respuestas y sitios similares. A esta calidad de respuestas, es natural que *Wikipedia*, donde se intenta asegurar una cierta veracidad mediante la

delegación de monitoreo a personas relevantes, parezca La Sorbona en comparación al "El Rincón del Vago".

El problema es cuando la respuesta no se encuentra, y hay que ayudar a construirla. O cuando se debe comenzar un hilo de discusión en foros especializados. Aquí el perfil tímido, propio de los informáticos, se convierte en parte del atasco en lugar de parte del problema. Ignorar comunidades de conocimiento construidas por y para los docentes de tecnología es solo la punta de hilo de un gran problema.

Para empezar, el docente de informática no sabe preguntar o no se anima. Los docentes subestiman incluso a los buscadores. A pesar de que todos ellos conocen a *Google*, es llamativo lo poco que lo usan. No se sorprenderían de otra manera, cuando se les demuestra que puede encontrar tutoriales en vídeos para encuadernar libros, aprender a tocar el violín, reparar canillas, armar cables, hacer *sushi* o compartir en tiempo real sus documentos con alumnos. No es de extrañar que se sorprendan también de los cientos de tutoriales que hay sobre su mismo campo de trabajo, o de la facilidad que supone reciclar material en aulas virtuales como *Moodle*. El *infrauso* que los propios docentes de tecnología e informática hacen de la misma tecnología es alarmante, y la respuesta es muy sencilla: *el docente de tecnología no pregunta en caso de atascos aparentemente "complejos", ni siquiera a Google. No pregunta simplemente porque no se imagina que la respuesta pudiera estar allí.*

Otro de sus temores es que la respuesta pudiera estar en otro idioma: al fin y al cabo el español (ya lo habíamos descrito en el capítulo 3.1.2) es solo el 17% de la web. Increíblemente, sus autismos suele ser tan palmarios que creen que su problema es único, y que a pesar de haber millones de usuarios que usan lo mismo, ellos son los únicos atascados.

#### 3.2.4. Las listas de élite

No es difícil de comprobar que, en los países latinos, los títulos llevan a una cierta susceptibilidad a quien los obtienen. Por tal razón, los profesionales prefieren un trato —si no diferencial—, que al menos no sea informal.

En el otro extremo del trato formal, están los foros sobre *software* libre que tienen a ser muy informales y especialmente meritocráticos, al punto que en estos foros los galardones impartidos por casas de estudio oficiales, prácticamente se esconden para no ofrecer una impresión equivocada a los *hackers*, empleándose esta expresión respecto del nivel de experticia que tienen estas personas en sus propios campos, que no tienen que ver estrictamente con seguridad ni con la vulneración de datos propuesta por el cine de Hollywood.

Paralelamente, se ha observado en reiteradas oportunidades que cuando los docentes de tecnología se atascan, no buscan la razón hasta sus orígenes. Tampoco buscan la solución al final de la clase, y vuelven a atascarse al año siguiente en el mismo problema, lo cual sugiere que tampoco han buscado la solución en sus ratos libres. Si se les ayuda en el momento, no toman nota, la rechazan por encontrarse delate de terceros o por el contrario, si el atasco ha ocurrido en clase, solicitan ayuda de malas maneras al encargado de turno. Sin embargo todo apunta a que no buscan la solución del problema, y que no lo hacen porque les cuesta imaginar completamente todas las aristas del problema; no le dan forma a la respuesta. Es decir, no se puede obtener una respuesta de la que al menos no se ha podido construir alguna de sus partes. Tampoco preguntan en foros, o cuando lo hacen, omiten toda clase de reglas de etiqueta de la red.

También hay un desaprovechamiento de la capacidad que tiene Google, propia de los *bot* (robot por *software*) de no impacientarse cuando la pregunta no

se realiza bien. De hecho Google frecuentemente encuentra mejores soluciones a preguntas realizadas en forma muy elemental que a preguntas planteada de forma compleja.

Pero cuando la respuesta no esta en Google, y debe ser planteada quizás por primera vez en un foro, son seres humanos quienes colaboran en la respuesta. Los seres humanos *sí son impacientes* y esperan al menos *un poco de colaboración* por parte del interesado. Por tal razón, muchos foros y listas de correos recomiendan a los nuevos participantes, que antes de preguntar, lean completamente el documento *¿Cómo hacer preguntas de manera inteligente?* de Raymond (2000) —un verdadero texto de culto del cual se citarán algunos párrafos—:

"Nunca asumas que tienes derecho a una respuesta. No lo tienes. Te ganarás una respuesta, si te la ganas haciendo una pregunta sustancial, interesante y que haga pensar, una que contribuya implícitamente a la experiencia de la comunidad antes que solicitar de manera pasiva conocimiento de los demás".

En otras palabras, las listas de correo poseen una suerte de elitismo necesario para poder mantenerse. Esperan al menos que uno demuestre que buscó la respuesta antes, que intentó algo y que explique dónde se quedó atascado. A veces los docentes temen que hayan arribado a un nivel tan alto de complejidad que nadie tenga la respuesta, por lo cual abandonan el intento, a pesar de que la realidad indica que están cerca de obtener ayuda especializada. Si se sigue el texto de Raymond:

"Lo primero que tienes que entender es que a los *hackers* les gustan los problemas realmente complejos y las buenas preguntas que les hagan pensar en ellos. De no ser así no estaríamos aquí. Si nos proporcionas una cuestión interesante, te estaremos agradecidos; las buenas preguntas suponen un estímulo y un regalo".

No obstante, los profesionales latinos son ciertamente susceptibles. Si no reciben respuestas, o si no las reciben en forma rápida o privada —ya que no les gusta exponerse—, entonces infieren que los foros y, especialmente aquellos de usuarios de *software* libre, están repletos de *enfants terribles*, insufribles y vanidosos *nerds*. Lo cual también es cierto; los *nerds*, además de contestar por solidaridad, también contestan por vanidad. Contestan para mejorar su nivel en estos espacios que como habíamos mencionado, son meritocráticos. Finalmente, contestan porque saben que la acción de ayuda a los novatos, les quita un peso a los "sabios", y les mejora las posibilidades de ser ayudados en preguntas realmente complejas.

La acidez de las respuestas es más alta cuando la pregunta es sobre *software* libre, y quien la plantea no se ha tomado la molestia de leer alguna introducción mínima sobre el tema. Por ejemplo en los foros se supone que todos saben que los comandos están documentados y que se autoexplican a la orden *man* (manual). Por ejemplo: *man firefox*, o *man unzip*. Por tal razón, la respuesta a cómo descomprimir un archivo *zip*, equivale a recibir de inmediato el conocido *RTFM*, expresión frecuente que indica que la respuesta es extremadamente fácil de encontrar, y cuya traducción literal no se efectuará en este texto. Un profesional susceptible que recibe este consejo y se lo toma en serio, normalmente no vuelve a preguntar y se retira airadamente del foro. Tampoco vuelve a intentar el experimento a fin de no volver a quedar atascado.

### **3.2.5. Impacto sobre los alumnos**

#### *3.2.5.1. Neofobia por inducción*

Argentina pasa por un proceso de tecnificación muy discutido en los últimos años, según el cual, en lugar de crear piezas en el país, se limita a importar las partes de China, ensamblarlas y ponerles una calcomanía "hecho en Tierra del Fuego". La dependencia de Asia es total.

Este proceso es análogo al de las aulas. Como ya fuera señalado, el alumno copia, pega y se adjudica lo encontrado. El docente de tecnología es como un capataz, un especialista en *marketing* que mide la belleza del *packaging* final. Sin embargo, si Internet se corta o la enciclopedia en CD se rompe, también se corta la clase. De acuerdo a lo indagado en la Escuela Normal 9-002 "Tomás Godoy Cruz", los docentes de tecnología no se acercan a la nutrida colección de textos técnicos de la biblioteca (hoy devenida en museo) a buscar lo mismo que estaba googleando minutos antes. En un torpe afán por proteger a los alumnos del conocimiento, éstos son ubicados en el papel de "indefensos usuarios" que se encuentran arrojados a lontananza si el encargado de laboratorio no corre a auxiliarlos. Todo lo que sea distinto a las herramientas de *Microsoft*, y que conlleve conceptos oscuros como enrutadores, protocolos y cables, toman el papel de entidades indeseables, que como si tuvieran conciencia propia, complotan contra la cadena de montaje. *Lo complejo es malvado; lo simple es bueno*. Las computadoras, al igual que los libros, se usan lo menos posible. Esta es la razón por la cual los alumnos son infrausuarios: porque infrausan la computadora de la misma manera que los libros: sólo buscan superficialmente los temas necesarios, nunca los leen completamente. Luego dependen tanto de los servicios técnicos como de los fabricantes asiáticos por igual.

En cierta manera existe un parecido con aquellos primeros homínidos que no podían conservar el fuego, que temían su naturaleza y que debían *comprarlo* a quienes conocían la técnica para crearlo mediante pedernal o frotación.

Hoy en día Argentina necesita alquilar puentes para poder llegar a otros lados. Necesita que otros países pongan sus satélites en órbita y que barcos extranjeros tiendan la fibra óptica bajo el mar. Los alumnos que salen del ciclo de nivel Medio poseen el mismo miedo a que ("se apague") la tecnología que portan. Con semejante respeto es natural que debamos pagar por la tecnología a países extranjeros. No importa cuanto cierren las fronteras a las importaciones: nuestros alumnos no comenzarán empresas de *hardware* si tienen miedo a que su propia computadora "se rompa con el uso".

Cuando un puente se corta, debería construirse otro mejor en su lugar. No se debería enseñar a cruzarlo tan sólo para llegar al otro lado, se debe, en su lugar, enseñar a construirlos, para que sean otros quienes puedan llegar a nuestra orilla.

### 3.2.5.2. Consumidores y productores

Una característica que comparten, tanto los dispositivos de *hardware* actuales, *tablets* y *celulares*, como los sistemas operativos Windows, Android y IOS, en relación a las viejas computadoras compuestas por MSDOS, es una previsión bastante completa de las necesidades del usuario. El usuario en general no necesita ir más lejos de sus límites habituales, y se puede concentrar en el contenido que se le ofrece. La necesidad de romper los límites impuestos por el *hardware* o por el *software* prácticamente no existen en la actualidad, en relación a las primeras PC.

Sin embargo la cantidad de producción de ciencia, de nuevos contenidos o de nuevas aplicaciones es generalmente limitada en relación a la cantidad de dispositivos tecnológicos con los que cuenta una familia tipo. En términos generales se cumple que el sujeto pase de poseer un objeto a ser poseído. La fascinación que ejercen estos dispositivos, juguetes sofisticados, son comparables al de las primeras muñecas francesas que describe Barthes (1957; 2009, pp. 52) en su *Mitologías*, donde refiere su preocupación respecto de aquellos juguetes *tan completos* que provocan en el niño la sensación de *no tener nada por completar*. He aquí el peligro del colapso de la imaginación y la creatividad, cuando todo está aparentemente resuelto. Esto quizá sienta las bases para otorgar un carácter tan real al "juguete" como a cualquiera de las cosas del mundo real.

"[...] ante este universo de objetos fieles y complicados, el niño se constituye, apenas, en propietario, en usuario, jamás en creador; no inventa el mundo, lo utiliza. Se le prepara gestos sin aventura, sin asombro y sin alegría. Se hace de él un pequeño propietario sin inquietudes, que ni siquiera tiene que inventar los resortes de la causalidad adulta; se los proporciona totalmente listos: sólo tiene que servirse, jamás tiene que lograr algo. Cualquier juego de construcción, mientras no sea demasiado refinado, implica un aprendizaje del mundo muy diferente: el niño no crea objetos significativos, le importa poco que tengan un nombre adulto; no ejerce un uso, sino una demiurgia: crea formas que andan, que dan vueltas, crea una vida, no una propiedad. Los objetos se conducen por sí mismos, ya no son una materia inerte y complicada en el hueco de la mano. Pero esto es poco frecuente: de ordinario, el juguete francés es un juguete de imitación, quiere hacer niños usuarios, no niños creadores".

El párrafo citado llama la atención sobre la motivación de las empresas que usan los dispositivos tecnológicos para convertir personas en usuarios, aprovechando el instinto cazador-recolector de las personas que creen poseer un objeto, para convertirse consumidores de la vanidad propia y ajena.

Hay algo inhumano en contemplar como millones de años de evolución tecnológica terminan coronados por la figura de seres humanos alienados, visitando perfil tras perfil de otras personas, ignorantes de la depresión que produce asomarse a un mundo que otras personas construyen como ideal, pero que es tan pequeño y humano como el de cualquiera. No es casualidad que desde hace años los psicólogos señalen que las redes sociales como *Facebook* provocan depresión en personas con baja autoestima o con identidades pobremente definidas.

Hay una manifiesta fuerza por parte de los docentes en apoyar sin reservas la adopción de las redes sociales, de subirse a tiempo, temiendo ser tachados de obsoletos y anticuados. Sin embargo, el docente debe ser criterioso en tal sentido; debe hacer propuestas formales de discusión, respecto de cuanta verdad hay en el *marketing* de los mundos personales. Debe legalizar el hecho que esas empresas intentan que las personas permanezcan *online* la mayor parte de tiempo posible para venderles publicidad. Debe llamar al respeto por el autoengaño presente en los perfiles ajenos. Hacer *click* en el "Me gusta" de actos inconscientes realizados por personas inmaduras, desequilibradas o con problemas emocionales, es apoyarlos en sus actos despersonalizantes y empujarlos aún más a la pérdida de sus identidades en el hacinamiento del espacio virtual. No se puede contribuir a la degradación de un ser humano sólo por diversión. Tal acción es la que realizan aquellos personas incapaces de empatía, es decir, el perfil de los psicópatas.

Otro aspecto tiene que ver con la reconstrucción de mundos posibles. La generación actual no sabe, por ejemplo que el consumo de cultura pasó de los blogs a las redes sociales, y que la distribución de los ingresos quedó concentrada en estos gigantes. No todo en Internet es evolución. Protocolos interesantes como RSS y Atom quedaron atrás para dar paso a los timelines de las redes sociales. La

producción de contenidos se desarmó frente al *streaming* de banalidades que pueblan las redes. Las buenas intenciones de los bloggers de vivir de la producción de contenidos quedó tapada por estos gigantes que mediante sus usuarios cautivos hurtan la producción ajena.

Lo interesante es que toda la tecnología que un estudiante necesita para crear una versión menos deshumanizada de *Facebook*, la tiene concentrada en el kilo y medio de su *netbook*. Los motores de bases de datos de estas grandes compañías, como MariaDB, PostgreSQL, MongoDB o Cassandra, pueden ser instalados en las netbooks con unos cuantas órdenes. Estos motores corren sobre las mismas versiones de *Linux* que tienen los estudiantes, tales como Ubuntu y Debian. Internet por su parte provee el espacio en hostings y clouds inicialmente gratuitos como Amazon WebServices, Openshift, DigitalOcean, Heroku, etc. Incluso el dinero necesario para empezar puede recaudarse en Kickstarter, Croudfounder, Somolend y otras entidades dedicadas al micromecenazgo.

Técnicamente el alumno, o el grupo de alumnos que compartan una PC, tienen lo necesario para comenzar la construcción de "mundos" (escenarios virtuales) en función de espacios auténticos de conocimiento humano. El límite se encuentra en el docente.

No obstante, si crear semillas de equivalentes a *Facebook* o *Twitter* parece una empresa muy grande para el trabajo del profesor, se debe señalar que con unos cuantos rudimentos de programación se puede construir aplicaciones en el aula que pueden ser embebidas adentro, tales como juegos educativos. Incluso *Facebook* mismo promueve el desarrollo de estas aplicaciones, es decir, mundos o esferas dentro de mundos.

Descendiendo la escala de complejidad, si el docente no quiere llevar a los alumnos al terreno de la programación, puede sugerir en cambio la creación de

páginas y de grupos dentro de la red social, tales que aprovechen el tráfico de la red para interesar a los navegantes a formar parte de actividades más dignas de nuestra civilización. Se beneficiará así de la masa de usuarios que ya se encuentran "logueados" en el sistema. Por supuesto, deberá competir con todas las noticias irrelevantes que se cuelan por todas las partes de la *interfase* que Facebook deja libre fuera de las publicidades.

Finalmente, si el docente no sabe como crear un grupo nuevo, puede sugerir como ejercicio para sus alumnos la participación en grupos donde se privilegien las emociones estéticas, la cultura, la diversidad o la ecología —sólo por nombrar algunos tópicos<sup>43</sup>. El docente puede hacer el seguimiento de las actividades del alumno en estos grupos (públicos), ayudarlo mientras participa, revisando la calidad de sus "posteos" y colaboraciones.

A propósito del tema, rescato las siguientes palabras de Jerome Bruner (2004, pp. 136), quien dice al respecto:

"Si no llega a desarrollar ningún sentido de lo que llamamos intervención reflexiva en el conocimiento que halla, el joven estará actuando continuamente desde afuera, el conocimiento lo controlará y lo guiará. Si logra desarrollar ese sentido, controlará y seleccionará el conocimiento según sus necesidades. Si desarrolla un sentido del *self* que se base en su capacidad de ahondar en el conocimiento para sus propios usos, y si puede compartir y negociar el resultado de sus profundizaciones, llegará a ser uno de los miembros de la comunidad creadora de cultura".

Llevado a nuestro contexto, la anteriores palabras trascienden lo mucho que se pierde ignorando las posibilidades que brinda una simple PC.

43 Anónimo (2016). *Un grupo de jóvenes "antiplástico" se moviliza por una Mendoza sustentable y limpia las calles*. Recuperado de <http://www.elsol.com.ar/nota/173232>, [consulta el 20 de agosto de 2016]

Para ilustrar la idea, conviene retrotraerse a la época en que Internet no era más que una costosa infraestructura y unas pantallas negras descifrables solo por usuarios avanzados. Todo montado mediante costosos *routers* y servidores Sparc provistos por SCO Unix, Solaris y Sun Microsystem. Un terreno limitado solo a universidades, empresas grandes o gobiernos, capaces de destinar grandes sumas de dinero.

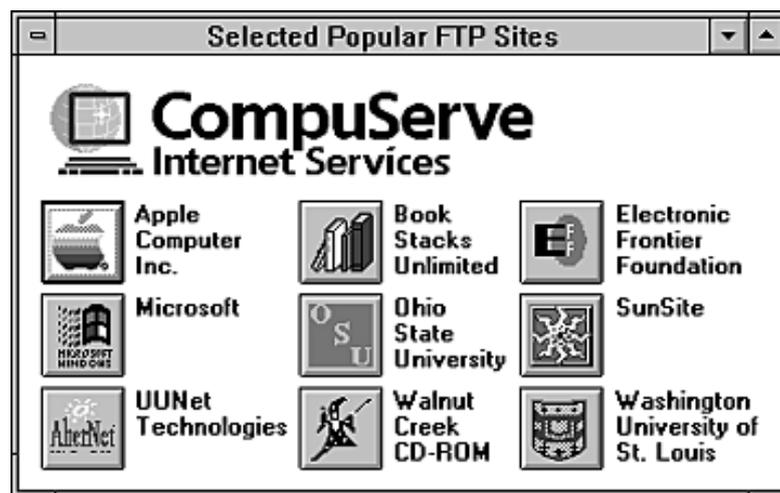


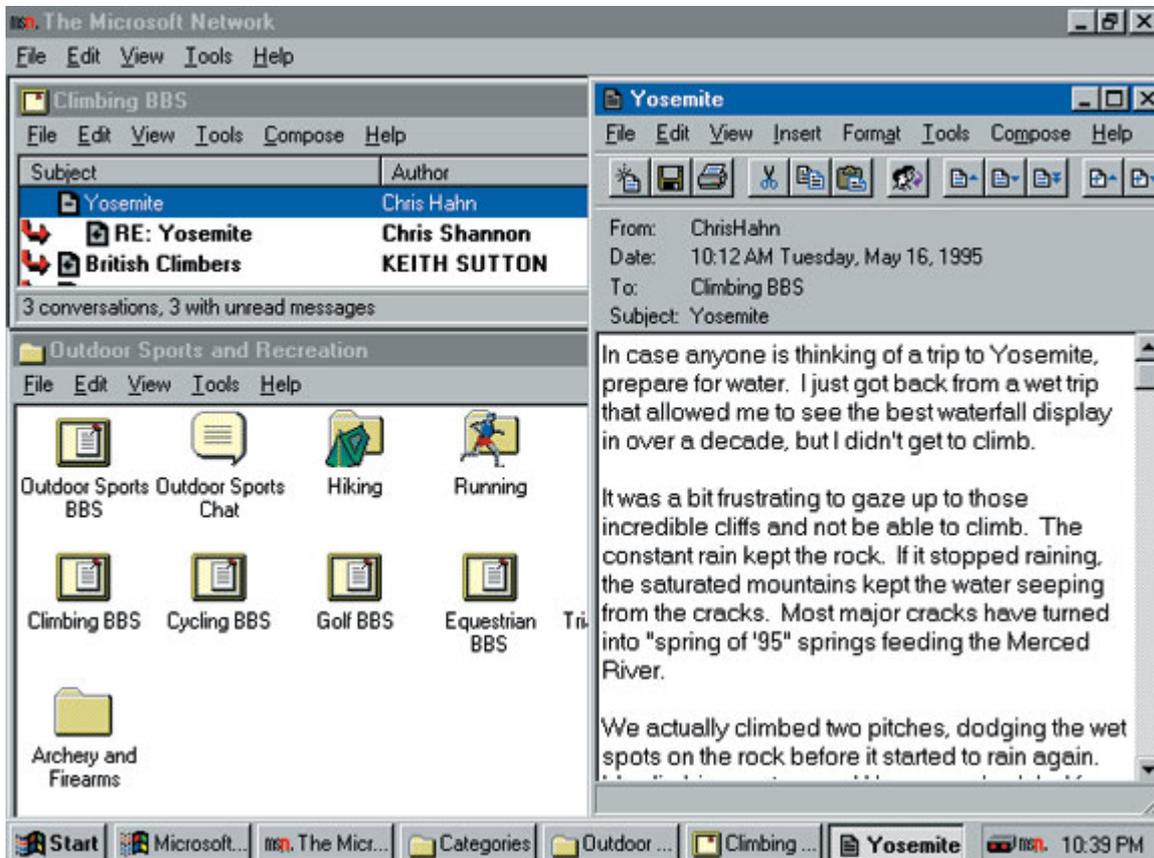
Ilustración F: CompuServe

Por entonces, las compañías CompuServe y The Microsoft Network ofrecían servicios equivalentes a BBS (Bulletin Board System), con *interfases* visuales un poco más fáciles de usar, por un abono que rondaba entre u\$s 35 a u\$s 60, extras a la conexión a Internet, la cual tenía un costo aproximado de u\$s 300 mensuales, sin contar los pulsos telefónicos. Ésto sin contar la inflación norteamericana a la fecha.

Hacia mediados de los años 90, llegó a su madurez GNU/Linux, un equivalente a Unix gratuito, que tenía tres particularidades, a saber;

- podía correr en PC normales,
- permitía ejecutar el programa servidor Web Apache,

- podía actuar de *router*



*Ilustración G: The Microsoft Network: la Internet "fácil" que no fue, debido a la llegada de la Web gratuita de Linux y Apache*

Las páginas HTML que se podían servir con Apache eran sencillas de programar, y a diferencia de Telnet (consolas remotas de caracteres) o FTP (protocolo de envío de archivos), eran del agrado de los usuarios por la sencilla razón que podían consumirlas sin conocimientos especiales. Rápidamente miles de pequeños servidores GNU/Linux comenzaron a conectarse ampliando Internet. Las pantallas fueron mejorando, y actividades como comunicarse (protocolo IRC) y leer mails (protocolos de correo POP, IMAP y SMTP) quedaban asequibles al público lego mediante su enmascaramiento con HTML y más tarde, Javascript.

Al cabo de unos pocos años, la inmensa cantidad de nodos disponibles provocó un aumento en la demanda, una consecuente abundancia de oferta, y una masiva bajada en los precios en las conexiones, hasta llegar a la actualidad en que Internet y la Web son económicamente accesibles, al nivel de servicios públicos como gas, agua o electricidad.

Pequeños avances a veces logran grandes cambios. Linux (Universidad de Helsinki) y sus partes, las GNU Tools (Instituto de Tecnología de Massachusetts), el protocolo TCP/IP (Universidad de Berkeley) han sido concebidas por estudiantes y profesores, aprovechando lo disponible, sin hacer caso a compañías comerciales. Valga la aclaración: acceder a Internet es gratis. Lo que el usuario paga es la conexión al ISP (Internet Server Provider). Si dependiera de estas compañías, unas pocas personas con dinero serían las únicas que seguirían consumiendo sus *interfases facilitadoras*, conectadas mediante rústicos modems, y obteniendo contenidos pre seleccionados por los *sysop*.

El docente es una figura clave. Es la persona más apropiada para mostrar a los alumnos la forma en que pueden pasar de consumidores a asumir roles activos en la sociedad creadora de cultura. Deconstruir la forma en que los nodos se conectan y se mantienen, al menos a grandes rasgos, les permite a los alumnos dejar de considerar a la Web como una esfera "irreductible", abstracta e incomprensible. Al menos durante unas clases, acciones simples como correr a un lado la amable cortina de ventanas, mostrarles comandos de MSDOS o de la terminal de Linux, representan cambios gigantes en cuanto a conceptos. La magia se devela para convertirse en elementos que el alumno puede utilizar, rechazar o aceptar, y en este último caso, seguir explorando en la universidad, con una idea mejor formada de lo que le espera en casos de seguir estudios de sistemas o

comunicaciones<sup>44</sup>, bastante más completa que las herramientas ofimáticas básicas que aprende en su obsoleto programa de estudios.

44 Este tema se profundizará mediante la propuesta de varias ideas para aplicar en el aula, en el Apéndice B.

## Capítulo 4: Propuesta metodológica

### 4.1. Enfoque de Investigación

Si bien originalmente se planteó un abordaje metodológico mixto, la recolección de datos pronto evidenció el problema de obtener datos cuantitativos fiables. Esto se debe a que las encuestas realizadas personalmente mostraron ciertas parcialidades (falseamiento de la información requerida) provocadas por la naturaleza misma del estudio. Preguntas situadas en el cuestionario más adelante *ex profeso* para triangular la respuesta, mostraron contradicciones que evidencian el temor en los docentes de informática a perder sus puestos de trabajo, sin tener una idea cabal de cuanto saben o están dispuestos a aprender. Las encuestas suministradas *on line* profundizaban esta disparidad de respuestas. Cuando no había obligación de responder, la encuesta era ignorada. La encuesta enviada, por ejemplo, a varios grupos de Facebook de aproximadamente por 8.000 miembros, solo fueron respondidas por apenas 12 personas.

En cambio, las entrevistas mostraron mucha información y líneas de investigación a seguir, de modo que los esfuerzos fueron orientados completamente al enfoque cualitativo, mediante un estudio de casos múltiples, a fin de librar de ambigüedades merced a la triangulación de perfiles y títulos profesionales, y a fin de identificar posibles patrones tanto en las trayectorias de formación, así como en el tipo de resistencias al cambio.

## 4.2. Método de Investigación

La presente investigación adoptará el método de casos múltiples descrito por Jordi Caïs (1997) en su Metodología del Análisis Comparativo. Se escoge este enfoque por el giro social que ha ido tomando la presente investigación, a medida que se ha ido desarrollando desde su ingenua concepción original.

Este enfoque, bajo su estrategia "estudio de casos", es adecuada para la cantidad de unidades de datos recolectadas, y permite efectuar un conjunto de comparaciones bajo un marco formal de estudio ya comprobado en el ámbito científico. La acción de efectuar comparaciones entre los casos obtenidos es también apoyada en la obra de Jordi Caïs: de acuerdo a sus palabras "toda investigación social implica algún tipo de comparación" (1997, pp. 11).

Se toma en cuenta asimismo los límites de este método, los cuales están determinados por las propias ciencias sociales. En ciencias sociales hay un ideal obtenido de las ciencias naturales, donde el ciclo de la experimentación permite llegar a predicciones fiables (observación, hipótesis, experimentación). Sin embargo en ciencias naturales se puede recrear entornos controlados donde aislar las unidades de estudio de tal forma que no se vean influenciadas por otras variables de entorno. Este mecanismo de *ceteris paribus* es inaplicable en sociología y en nuestro estudio.

Sin embargo, de acuerdo al autor, de utilizar el método de estudio de casos que comparten ambas ciencias, se pueden aplicar "generalizaciones modestas" capaces de "aproximarse al rigor experimental mediante la identificación de efectos comparables de un fenómeno, y el análisis de similitudes y diferencias entre ellos. Este punto de vista provee las bases para establecer generalizaciones empíricas concernientes a categorías de fenómenos sociales históricos" (1997, pp. 18).

El análisis parte de trayectorias de vida académicas y profesionales construidas mediante entrevistas en profundidad a informantes clave, cuya formación profesional se encuentra enmarcada en ingeniería en sistemas, licenciatura en informática, análisis de sistemas, técnicos universitarios y profesorado en tecnología. En caso de encontrar patrones en aquellas respuestas extensivas y de poseer estas unidades de información significaciones equivalentes, se utilizará las recurrencias para la identificación de patrones por similitud. Ello permitirá ulteriormente apoyar o rechazar los intentos de explicación.

### **4.3. Población y tamaño de muestra**

El muestreo para la entrevista es de tipo no probabilístico, basado en una población de 15 sujetos tipo.

Respecto de este número, se pone énfasis en la profundidad de los datos obtenidos antes que en la cantidad. De acuerdo a Jordi Caïs, "pocas veces se puede utilizar una variedad de casos que sea suficiente para probar o negar argumentos causales. El número (siempre) limitado de casos impone una cierta indeterminación. El investigador, sin embargo, puede dar más énfasis a sus explicaciones citando circunstancias periféricas. Esta atención a los detalles de los casos individuales produce un diálogo de investigación fructífero entre el investigador y la evidencia" (1997, pp. 18).

Respecto de los sujetos tipo a considerar, éstos se han de desempeñar en alguno de los siguientes cargos:

- MEP (Maestro de Enseñanza Práctica) de Informática;
- ATP (Auxiliar de Trabajos Prácticos) de Informática;
- AIE (Auxiliar de Informática Educativa);
- docentes de Tecnología; y

- docentes de Educación Tecnológica.

A tal efecto se realizarán 15 entrevistas en profundidad, a saber: a) 3 ingenieros en sistemas o en informática; b) 3 licenciados en sistemas o en tecnología educativa; c) 3 analistas en sistemas; d) 3 profesores de tecnología o de informática; y e) 3 otros títulos habilitantes, tales como técnicos en redes o diseñadores industriales. Estas entrevistas permitieron relevar información acerca de las trayectorias académicas de formación, las motivaciones de estudio, las trayectorias profesionales y las representaciones sociales que tienen de sí mismo los docentes. También se incluyeron otras preguntas para tanto distraer al entrevistado como para obtener algunos otros rasgos de su personalidad u ocupaciones.

Además se tomaron en consideración a lo largo de toda la obra otras variables teóricas planteadas en investigaciones sobre teorías del cambio, procedentes de autores como Paul Watzlawick, John Weakland y Richard Fisch, y sus colegas investigadores del *Mental Research Institute* de Palo Alto: Gregory Bateson (quien trabajó holismo, sistemas y cibernética) y Donald D. Jackson (dedicado a profundizar respecto de cómo la evolución de los sistemas homeostáticos, desde un punto de vista familiar, pueden mejorar si son capaces de incluir elementos nuevos). También se apelaron a los resultados planteados por Heinz von Foerster y Norbert Wiener (por sus contribuciones a la cibernética de segundo orden).

## **4.4. Instrumento de recolección de datos**

### **4.4.1. Entrevistas en profundidad**

Puesto que la población objeto de investigación es muy vasta, el muestreo será de tipo no probabilístico según sujetos tipo (en función de las carreras de formación profesional y las trayectorias profesionales de los mismos).

Las entrevistas buscan relevar datos significativos de informantes clave para poder analizar las trayectorias de vida y su impacto en su desarrollo profesional. A tal efecto los individuos muestreados serán seleccionados no probabilísticamente, según sujetos tipo y sujetos voluntarios. Los sujetos considerados serán profesores de tecnología, analistas en sistemas, técnicos en redes, técnicos programadores, ingenieros en informática, ingenieros en sistemas, licenciados en sistemas o diseñadores industriales, que actualmente se desempeñen como MEP (Maestro de Enseñanza Práctica) de Informática, ATP (Auxiliar de Trabajos Prácticos) de Informática, AIE (Auxiliar de Informática Educativa) y docentes de Tecnología, tal cual se indicó en puntos precedentes.

El formato de la entrevista, si bien sigue una secuencia preestablecida, se realizarán de manera no fuertemente estructurada, con la idea de lograr establecer un vínculo de confianza en los interlocutores y de profundizar lo más posible en las auténticas motivaciones y razones que subyacen en cada respuesta.

### **4.4.2. Fichaje documental**

El proceso de fichaje partirá de la identificación de palabras clave sobre el tema de la investigación y la posterior identificación de gran hitos teóricos propios del campo y área de investigación. La identificación de documentos se iniciará a través de la búsqueda de documentos bajo el formato de libros (tanto en soporte

papel como en soporte digital), seguido por la búsqueda en bases de datos y en bibliotecas virtuales de material hemerográfico actualizado sobre el tema, así como aquellos recomendados por el Director de la Tesina.

Una vez realizada la identificación a través de un fichaje descriptivo, cuyos documentos serán catalogados según referencias de las normas de la *American Psychological Association*, sexta edición. Seguidamente se procederá a efectuar las fichas de estudio relevando citas textuales, registro de ideas generales y datos precisos de importancia con la consignación de sus datos referenciales. Estos registros irán acompañados de notas y comentarios acerca de conexiones con otros datos, variables y categorías que emerjan al construir el objeto de estudio, las definiciones teóricas y operacionales, y el análisis lógico argumental consecuente. Ello posteriormente será triangulado con los datos empíricos recabados, a fin de poner a prueba los supuestos de partida (hipótesis) consideradas a modo de contrastación empírica de las ideaciones conceptuales o constructos de análisis propuestos.

Una vez organizados los datos documentales y clasificados temporalmente se procederá a la lectura y elaboración de fichas de estudio —como se indicó *supra*—, que darán lugar al estado del arte, organizando y construyendo el estudio de los antecedentes y del marco teórico con sus descripciones, clasificaciones y definiciones teóricas y operacionales indispensables para el abordaje de la base empírica en cuestión a través de la zona teórica mencionada.

#### **4.5. Análisis de la Información: Planteo Teórico**

En este punto se emplea la expresión *Información* en lugar de *Datos*, por la razón que las unidades de información obtenidas tienen significancia por sí mismas.

Se realizará una triangulación de unidades de información dentro de las entrevistas para detectar si el docente dudaba, mentía o aún no contaba con suficientes claridades conceptuales, procedimentales y pragmáticas. La idea consiste en identificar autojustificaciones que resultarían recurrentes entre los docentes de Tecnología y los encargados de laboratorio, del tipo "no asistí a las capacitaciones por realizarse estas fuera de mi horario de cursado", "no tengo tiempo", "no pienso cambiar los apuntes de mis estudiantes" o "yo enseño lo que usa 'todo el mundo'".

Como se planteó anteriormente, se tomará en cuenta si las respuestas aparecen más de una vez, tabulando de acuerdo con la frecuencia con la que aparecen (tendencia).

El diseño exploratorio secuencial DEXPLOS, descrito por Hernández Sampieri, Fernández Collado y Baptista Lucio (2010, pp. 564), es el que se muestra más adecuado para encontrar patrones en las unidades de información recolectadas. Este diseño parte de los datos obtenidos en forma cualitativa, para trabajar sobre ellos en forma cuantitativa.

Respecto de la forma de tratar las unidades prospectadas en la documentación disponible, se armó inicialmente una red conceptual que relacione conceptos, autores y teorías mediante CmapTools, de tal modo que permita sugerir varias líneas de acción. Sin embargo, tal triangulación teórica con los testimonios surgidos de los datos cualitativos, exige comparar una gran cantidad de cadenas de texto emergentes de las entrevistas, para compararlas con la bibliografía del marco teórico.

Por lo tanto, la investigación empleará *softwares* disponibles de tipo CAQDAS (*Computer Assisted Qualitative Data Analysis Software*). Luego de varios análisis y pruebas con cuatro opciones del mercado, entre los cuales se puede mencionar a RQDA, CATMA y QDA Miner, se optó por utilizar ATLAS.ti. A fin de alimentar el motor de texto se procederá a cargarlo con el material que se dispone en su versión digital. Para los casos en que solo se dispone de la versión en papel, se procederá a identificar varias unidades de información pertinentes. Estas unidades se han transcrito e inserto en el programa para ser incluidas en el análisis. Ello enriquecerá notablemente la correlación de distintas categorías de análisis, que facilitarán notablemente la construcción de un potencial modelo teórico asociado al tema.

## Capítulo 5: Análisis de la información: personal de laboratorio, docentes de informática y tecnología en acción

Para la siguiente descripción se parte de las preguntas iniciales de investigación. A partir de allí se ponen en consideración las hipótesis o supuestos de partida y sus correspondientes resultados, en forma de progresión discursiva.

Durante la tabulación de datos se encontró que algunos de los sujetos entrevistados tenían más de un título habilitante, por lo que se tiene en cuenta para detectar tendencias. Con el mismo objeto, y a pesar de tener todos los casos necesarios (3 de cada categoría, 5 categorías: A, L, I, O, P) también se tabulan sujetos que se presentaron espontáneamente a ser analizados.

N.º Caso	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Orden DP en Atlas.ti (Revisar Anexo 2 en CD)	P1	P8 P9	P4	P2	P6	P17	P12	P13	P18	P11	P49	P14	P5	P15	P10	P3
A- Analista en Sistemas																
L – Licenciado en Sist.																
I- Ingeniero																
O <sub>T</sub> - Técnico en Redes																
O <sub>D</sub> - Diseñador Industrial																
O <sub>P</sub> - Programador																
O <sub>tu</sub> – Técnico Universit.																
P- Profesor en Informática																
P <sub>T</sub> - Profesor en Tecnolog.																
P <sub>F</sub> - Profesor en Filosofía																

La primer pregunta [P1], relativa a los estudiantes y egresados de carreras de TIC, plantea si la resistencia al cambio es acaso un problema principalmente actitudinal, producto de un engaño primigenio, tanto inducido como autosostenido.

Respecto de que sea inducido, la indagación comienza formulando la siguiente propuesta: *Las instituciones de formación universitario o no universitaria establecerían claramente las actitudes y aptitudes que se requieren tanto de un estudiante como de un futuro profesional del área o del campo al que se va a dedicar* [H<sub>1.1</sub>]. Esta hipótesis indaga bajo una dimensión que delimite la claridad de las advertencias ofrecidas a los aspirantes respecto del esfuerzo psíquico a emplear durante y después de la carrera.

También se buscan rastros respecto que las advertencias podrían haber sido ignoradas *ex profeso*, —se indica supra (1.1.2. *El perfil equivocado: la habilitación de cada título*).

A fin de medir la validez (o invalidez) de la primera aseveración se han establecido los siguientes indicadores: a) distinciones efectuadas a los ingresantes respecto de las tareas que habilita cada título profesionalizante; b) conocimientos que debieron acreditar en un Pre Universitario; y c) noción de la dificultad que iban a encontrar diariamente en la vida profesional (advertencias).

Si bien la causalidad es un concepto metafísico (Aristóteles; Kant; Jordi Caïs, 1997, pp. 31), y no es la primera vez que señalaremos este hecho, aquí se indaga la relación de estos indicadores de acuerdo a su influencia sobre la probabilidad que el aspirante inicie o no sus estudios (en adelante: variable directa IE), y de esta manera examinar si la aseveración de la hipótesis es válida o no. En tal sentido una clave analítica reside en el primer mes de estudio en su carrera, a partir de lo cual cabe plantear dos vías indagatorias: una de tipo binaria y otra modal-relacional. Respecto de la indagación binaria, ante el interrogante simple "¿tuvo un Pre Universitario formal? [PRE]", es posible dos respuestas discretas, con solo dos valores: X (presencia) o 0 (ausencia); en cambio, su profundización, "Dificultad en el Pre Universitario o Curso Confrontación Vocacional y Nivelatorio"

[DPU] se abre un abanico en principio indeterminado (medir dificultad), pero determinable desde la restricción de los sujetos muestreados dentro de una muestra poblacional. Para esta segunda incógnita también se utilizará una variable indirecta, discreta, pero capaz de tomar más valores que los booleanos empleados anteriormente. Su continuidad se puede dirimir a la manera de Reynoso (2011, pp. 25), es decir, mediante la adopción de una escala, para luego definir su precisión de la siguiente forma: 0) *"Pre Universitario, Confrontación Vocacional y/o Nivelatorio no fue representativo"*, "estuvo ausente" o "no dejó ni una huella, no me acuerdo de nada"; 1) "tuve un Pre Universitario o una introducción liviana, con pocos ejemplos"; y 2) "el Pre Universitario o la introducción que tuve fue muy completa, con ejemplos de dificultad inicial comparable a la del resto de mis estudios, con ejemplos de lo que después fue mi vida profesional".

Para el caso de la variable directa [IE], que es "inician estudios", no se formula una sumatoria de las escalas anteriores. Por un lado, la hipótesis es de tipo descriptiva, de modo que no se restringe a una fórmula que combine variables. A su vez, de los entrevistados se sabe que terminaron de todos modos sus estudios —presencia determinada con una X—, porque la muestra poblacional *solo incluye egresados que hayan terminado sus estudios*.

La grilla anterior, una vez completada, queda conformada de la siguiente forma:

N.º Caso		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
Orden DP en Atlas.ti (Revisar Anexo 2 CD)		P1	P8 P9	P4	P2	P6	P17	P12	P13	P18	P11	P49	P14	P5	P15	P10	P3	Prom
A- Analista en Sist.	X <sub>1</sub> .PRE	X	X	X										O			X	
	X <sub>2</sub> .DPU	1	0	1										0			2	0,80
L – Licenc. en Sist.	X <sub>1</sub> .PRE				X	O	O									X		
	X <sub>2</sub> .DPU				1	0	0									1		0,50
I- Ingeniero	X <sub>1</sub> .PRE							O	X	X								
	X <sub>2</sub> .DPU							0	1	2								1
O <sub>T</sub> - Téc. en Redes	X <sub>1</sub> .PRE													X				
	X <sub>2</sub> .DPU													2				
O <sub>D</sub> - Diseñador Ind	X <sub>1</sub> .PRE										X							
	X <sub>2</sub> .DPU										2							2
O <sub>P</sub> - Programador	X <sub>1</sub> .PRE																	
	X <sub>2</sub> .DPU																	
O <sub>tu</sub> – Téc. Univers.	X <sub>1</sub> .PRE											X						
	X <sub>2</sub> .DPU											2						
P- Prof. en Inform.	X <sub>1</sub> .PRE													X				
	X <sub>2</sub> .DPU													0				
P <sub>T</sub> - Prof. en Tecn.	X <sub>1</sub> .PRE														X			
	X <sub>2</sub> .DPU														2			1
P <sub>F</sub> - Prof. en Filosof	X <sub>1</sub> .PRE															X		
	X <sub>2</sub> .DPU															1		
Promedio																		1,06

Como nota accesoria al proceso de análisis de datos se señalan a continuación algunos casos. Estos condujeron a la formulación de supuestos emergentes, es decir, hipótesis descriptivas que surgieron, tras una primera lectura de los datos relevados. En los casos de los Analistas en Sistemas y de los Licenciados en Sistemas, con excepción del caso P3, no hubo indicios de que apuntaran a un Pre Universitario que haya resultado significativo. Ahora bien, en el

caso P5 se muestra contundente al respecto, puesto que la informante clave se recibió en el Profesorado, pero empezó como Analista y pasó también por la Licenciatura:

B: —“En la primera carrera, no se realizó Pre Universitario. En la Licenciatura fue más una ambientación [...]. La ambientación fue para conocer las aulas, los laboratorios, la forma de cursar y todo lo referido a la aprobación y demás cosas referidas a como seguir la carrera. No me explicaron la diferencia de roles entre lo que hace el egresado de cada título.” (cf. Anexo 1: Entrevista P5).

Otra unidad de análisis interesante es el caso P2:

B: —[...] En el pre no nos marcaba la diferencia de la carrera, no nos mostraban los pilares de la carrera. No había programación, no había *hardware*, nada de eso. Era matemática. Me acuerdo que vimos funciones, conjuntos, bastante básica, Inglés básico, y otras dos materias [sigue sin recordarlas, y curiosamente no menciona algoritmos ni lógica].

El caso P2 también agrega algo que más tarde dará lugar a otra hipótesis, la [H<sub>1.2</sub>], al indicar lo siguiente:

A: —Y formalmente, ¿les hicieron la distinción respecto de que los licenciados, generalmente no programan, sino que dirigen proyectos?  
¿Les hicieron esa segmentación?

B: —[...]

A: —¿O les dijeron que "con este título van a poder hacer de todo"?

B: —Mirá... algunos sí te lo marcan; y otros no.

A: —Pero los profesores, ¿te lo marcan?

B: —Los profesores, los profesores —asiente con la cabeza.

A: —Yo te digo en el pre.

B: —Ah, ¿en el pre?

A: —En el pre.

B: —No, yo no tuve pre.

A: —¿Y exámenes pre ingreso?

B: —Hubo un... pre nivelatorio. Nos daban Matemática, Inglés básico y otra materia más

(cf. Anexo 1, Entrevista P2, pregunta 5).

Respecto de la omisión de los roles, o al menos aquellos típicos a cumplir con el título, se supone que estos han quedado claros con el folleto entregado en la Feria de la Oferta Educativa.

De acuerdo a los recuerdos manifestados por el Profesor y Licenciado de la entrevista P10 (cf. Anexo 1, Entrevista P10, pregunta 5):

A: —¿Eso fue lo que pensaste en aquél momento al elegir la carrera?, ¿por la salida laboral?

B: —Exactamente, no era que vos decís "me gustaba". No tenía ni idea de qué se trataba.

(cf. Anexo 1: Entrevista P10, pregunta 6).

Inicialmente se asignaría un  $X_2 = 0$  a esta aseveración. Sin embargo, el entrevistado concede un cierto valor a su Pre Universitario, una honestidad por parte de su institución educativa, cuando manifiesta:

A: —¿Cómo te presentaron la carrera?

B: —[...] el *marketing* de las carreras no se hacía como se hace ahora, que ahora van, te presentan la carrera, te hacen toda una presentación, un *marketing*.

(cf. Anexo 1: Entrevista P10, pregunta 5).

La necesidad de agregar una nueva hipótesis al tratamiento explícito de los roles se hace evidente cuando se encuentran otras coincidencias, tales como las siguientes:

B: —[...] te lo voy a decir de onda... muchos se metieron a la facultad creyendo que van a ser Operadores de PC. ¡Ojo! Yo tuve chicas, chicos en primer año que se pensaban que iban a aprender a manejar un Office, un Excel...

A: —Es decir, ¿qué podrían haber cursado un Operador en PC tranquilamente en vez de...?

B: —Exactamente... (cf. Anexo 1: Entrevista P1, pregunta 5).

A partir del análisis de este discurso se detecta evidencia que sustenta lo indicado en el capítulo 1, subtítulo 1.1.2. *El perfil equivocado: la habilitación para cada título.*

No obstante, es evidente que la universidad no es enteramente responsable de advertir a los aspirantes, y son las mismas materias de primer año quienes poseen una carga de complejidad suficientemente disuasoria. Hay entonces una visión idealizada de sí mismos, cuando a pesar de conocer a grandes rasgos la dificultad que pueden llegar a encontrarse, igualmente continúan y terminan sus estudios. Ello se patentiza gracias a una serie de extractos tales como esta aseveración de un Licenciado en Sistemas:

A: —Y en aquel momento, que me decís no fue fácil. Algo te llevó a estudiar sistemas. ¿Cómo te presentaron la carrera en la UCA, en el curso de nivelación o Pre Universitario?

B: —Quizás fue más hablar con mis primos y algunas experiencias... que te cuentan lo que es... Y bueno... me llamó mucho la atención.

A: —Previo a acceder a esa computadora, o incluso antes de estudiar, ¿qué acceso tenías a las computadoras o juegos?

B: —Y tenía... No, no. Era jugar. Yo incluso ya había hecho un curso cuando tenía doce años [acerca] de programación. Era como que venía conmigo. Tenía un chip y me llamaba la atención. Pero si ahora me preguntás si vuelvo a estudiar esta carrera, [diría] "no" —risas.

A: —¿Por qué?

B: —Y... principalmente porque es una carrera que te consume mucho la cabeza...

(cf. Anexo 1: Entrevista P6, pregunta 5).

Aquí se puede apreciar que no se tiene en cuenta el Pre Universitario, sino las opiniones de otras figuras en las que el estudiante confiaba, lo cual no debería ser una sorpresa: la autoridad es una fuente de conocimiento así como una base de una de las falacias más frecuentes (apelación a la autoridad o *argumentum ad verecundiam*), de acuerdo a las primeras advertencias lógicas o epistemológicas (Copi, 1973, pp. 69; Comesaña, 2001, pp. 60 y s.).

Otro aspecto a señalar es el detalle que hace mención a que, durante la carrera, pudo haber deducido la dificultad a enfrentar en el futuro, pero aun así la terminó, aunque después se arrepintió.

En el Extracto 3 aparece más evidencia de la inmadurez en cuanto a la elección de carrera, cuando un Técnico en Redes, "que venía de una ingeniería", patentiza su confusión al distinguir títulos, contenidos y competencias:

B: —Yo elegí sin saber lo que era Ingeniería en Sistemas, en la UTN, por ahí si vas a la carrera de la de Mendoza, ¿cómo es el título que te dan? Eh... Licenciado en Computación o una cosa así. Ahí está un poquito más focalizado, en cambio en la UTN es muy... lo que es la Ingeniería en Sistemas, es muy teórico, y es más referido a los sistemas en general. Te dan muchísimo análisis de sistemas. Te recibís de Analista en Sistemas primero, pero análisis de sistemas en general, no es un tema de Informática. Más que nada analizar una empresa, analizar un sistema, cómo funciona y nada más, o sea, no es tan referido a la Informática, y justo salieron estas carreras del ITU y me gustó la de Redes y Telecomunicaciones, y la verdad me gustó mucho. Estaba buena.

A: —¿Te acercaste a buscar información o te la presentaron a la tecnicatura? Tomando esto que decís, que suele pasar, "elegí Ingeniería en Sistemas" sin saber verdaderamente de qué se trataba...

B: — Claro, el enfoque que le dan en la UTN es un enfoque muy teórico, muy... No sé. Como que no está con la realidad informática, para mí está muy desfasado. No sé ahora si lo han cambiado, si han cambiado las materias, pero te dan Física, te dan Química, cuando vos decís, o sea, te pueden dar una materia así pero no tan profunda como la daban, y en realidad no tocaba materias referidas a la informática, programación no te daban nada. Te daban a principio de año Algoritmos, después más adelante te daban un par de cosas que te dan. Te machacaban con cosas viejas que ya no se usan. ¿Me entendés? Está muy mal armada para mí esa carrera de Ingeniería en Sistemas, por eso la gente que se mete ahí se termina cansando. Muchos se reciben porque bueno empezaron y la terminan, pero no te da las herramientas para poder... o la orientación está mal definida. No sé.

A: — ¿Y cómo fue tu experiencia en el ITU?

B: — En el ITU me gustó mucho más. Era muy práctico. Vos tenías teoría y ahí no más práctica. [...]

(cf. Anexo 1, Entrevista P14, pregunta 5).

Lo arriba expuesto pone de manifiesto una falta de madurez al inicio de su carrera de Ingeniería. Se queja de tener "sistemas..." en una carrera de Sistemas. Omite información sobre el Pre Universitario y se detecta que ni siquiera revisó el plan de estudios antes de comenzar. Se queja que tuvo Algoritmos, como una

materia desfasada, cuando en realidad no tiene por qué ser una materia "moderna" en referencia a los lenguajes de moda: recursividad, árboles, clases y objetos se pueden estudiar en el pizarrón, incluso de forma mucho más eficiente que enfocado en la computadora, bajo un lenguaje específico, lo cual cercena la capacidad futura de elección del estudiante.

Por otro lado, incluso si la universidad lo hubiera advertido adecuadamente, igual hubiera ingresado. Esto se detecta en que la segunda vez, eligió una carrera en la que se sentía más cómodo, solo por tener esta menos carga teórica. También menciona que los alumnos porfían en terminar la carrera incluso desalentados, lo cual coincide con el análisis realizado en el capítulo 1.2. *El alumno desapasionado a mitad de carrera.*

Antes de continuar explorando, se intenta medir la validez de la hipótesis inicial: H<sub>1.1</sub>, Las instituciones de formación universitario o no universitaria establecerían claramente las actitudes y aptitudes que se requieren tanto de un estudiante como de un futuro profesional del área o del campo al que se va a dedicar.

Si se considera el promedio general de los valores prospectados, el promedio llega a 1,06 (ingenieros y profesores), es decir, entra apenas en la categoría "tuve un Pre Universitario o una introducción liviana, con pocos ejemplos".

No obstante, este valor se encuentra atado al valor de Analistas y Licenciados, quienes al ser considerados en promedio dan lugar a un 0,80 (Analistas) y un 0,50 (Licenciados), cuyos rangos no llegan al valor de 1, con lo cual se los ha de ubicar en la primera categoría, a saber: 0) "la introducción / el Pre Universitario no fue representativo", "estuvo ausente" o "no dejó ni una huella, no me acuerdo de nada".

En las demás carreras, el promedio llega a 1 (ingenieros y profesores), es decir, "tuve un Pre Universitario o una introducción liviana, con pocos ejemplos", y llega a 2 en los Técnicos y Diseñadores Industriales, que sostienen que "el Pre Universitario o la introducción que tuve fue muy completa, con ejemplos de dificultad inicial comparable al resto de mis estudios, con ejemplos de lo que después fue mi vida profesional".

Para finalizar la contrastación de esta hipótesis hay que considerar que la misma ha sido comprobada en función de la base empírica prospectada, pero sujeta a los límites de la población considerada. Mediante el motor QDA y Atlas.ti se ha hallado mayor evidencia para construir y sustentar la segunda hipótesis: "De contar con el suficiente razonamiento lógico-matemático, o al menos una suficiente fuerza de voluntad, los alumnos de las carreras tecnológicas tenderían a finalizar sus estudios, incluso si han perdido toda la pasión que los ha motivado a emprenderla." [H<sub>1.2</sub>].

En este caso, la operacionalización de la hipótesis se ha examinado a partir de egresados como informantes clave, según los siguientes indicadores: 1) decepción durante el ejercicio de la profesión; 2) el ejercicio de otra actividad; y 3) la docencia exclusiva en tecnología (pues rehuye ejercer en empresas), siendo este último indicador no significativo, pero auxiliar para casos específicos, y no aplica al grupo de los docentes, puesto que a diferencia de las otras profesiones, la suya se ejerce específicamente en el aula.

Al revestir un carácter explicativo, la hipótesis admite una posible descomposición analítica en las siguientes variables:

- Aptitudes Necesarias (razonamiento lógico-matemático o voluntad suficiente) [AN];

- Componente Motivacional (pasión que ha motivado a emprender la carrera) [CM]; y
- Terminalidad Exitosa (alumnos que finalizan sus estudios y ejercen la profesión) [TEEP].

Si se aspira a una generalización y formalización matematizante, tales variables pueden presentarse de la siguiente manera:

$$(AN \cdot CM) \vee (AN \cdot \neg CM) \rightarrow TEEP$$

Respecto de AN, los entrevistados poseen al menos una de ambas condiciones (razonamiento lógico-matemático y/o voluntad), puesto que son *condiciones sine quae non* para terminar sus estudios: son todos egresados. En palabras de un ingeniero:

B: —Por ejemplo, [en] la UTN, te hacen mucho énfasis en los primeros años, en todo lo que es matemáticas, las materias duras.

A: —¿Las materias formales?

B: —Claro, si no tenés una predisposición para estudiar o no te la hacés en los primeros años, los primeros años son de adaptación, si vos no te adaptás en esos años, tenes que ser una persona, que sé yo, capaz de estar estos años, pero te adaptas, tiene que ser una persona que te adaptas, no así nomas y a abandonar todo, con esa actitud ya no tenes futuro.

A: — Digamos "voluntad".

B: — Voluntad, perseverancia, te adaptas, porque te tenes que adaptar, te tenes que acostumbrar a un montón de cosas, hay cosas que por ahí no te gustan, que las tenes que tarde o temprano las tenes que hacer tuyas, las

tenes que incorporar, y sobre todo tenes que empezar a tratar de resolver problemas... (cf. Anexo 1, Entrevista P12, pregunta 12).

Respecto a TE ["Terminan Estudios" de TEEP] todos concluyeron sus estudios, porque la muestra solo incluye egresados. La parte medible de TEEP es EP "Ejercen la Profesión", lo cual es una elección consciente y a sabiendas de la dificultad a esperar en el rubro. Es decir, mientras muchos egresados "cuelgan" el título y lo dejan como plan B, otros intentan el ejercicio de la profesión, tienen experiencias poco satisfactorias en el ámbito privado (*freelancer* o empresarial), y "caen" a las escuelas como refugio, donde pueden al fin (o no) dignificar su profesión.

Resta la variable CM, la cual se puede medir al indagar en el QDA por testimonios respecto de las razones por las cuales el entrevistado se encuentra dando clases, lo cual mide su grado de satisfacción respecto de lo que esperaba que fuera su vida profesional.

Para esto se aprovecha la presencia de preguntas abiertas realizadas durante la entrevista, que apuntan a lo siguiente: a) detectar rechazos en las empresas; b) determinar si hay baja autoestima en el docente respecto de lo que pudiera enfrentarse fuera de la escuela; c) indagar si surge una vocación docente adquirida con el tiempo, especialmente demostrable cuando emprende cursos de pedagogía, o carreras donde aprenda a enseñar más y mejor; d) Motivación aumentada o disminuida por el cambio tecnológico; y e) ventajas de trabajar en las escuelas, lo cual supone poner en valor el hecho de contar con un marco *formal*, *social* y *libre* respecto de los horarios de las empresas, del servicio técnico y del mercado informal de los *freelancers* a distancia.

Este último punto exige una aclaración por comparación. A diferencia de los *freelancers*, el armado de un horario semanal de horas, si bien exige reiteradas visitas a las escuelas, se recompensa con vacaciones, licencias, sindicato, jubilación en el estado y, lo que probablemente sea el punto más relacionado con el objeto de estudio de este trabajo, el grado de acceso a la complejidad por parte del docente (esto es ambiente técnico simple, poca necesidad de actualización y la posibilidad de un refugio mientras se sigue o no aprendiendo).

Este planteamiento se hizo fuera de las entrevistas y debería haber sido documentada, puesto que a guisa de agradecimiento a los entrevistados, se le ofreció a siete de ellos la posibilidad de introducirse en el mercado *freelance*. Tras examinar la especificidad de las ofertas existentes en portales tales como Upwork, Freelancer, Odesk, Workana, el vernáculo Trabajo Freelance, solo uno de los entrevistados se vio interesado en contar con una apoyatura en *Ruby* y despliegue de *CMS (Wordpress)* como productos extras a su oferta habitual de servicio técnico.

Si es docente o el encargado de laboratorio se encuentra satisfecho en su lugar de trabajo, el Componente Motivacional debe ser positivo. Para tener una idea de su grado de motivación, hacen falta comparaciones con otras poblaciones de estudio no consideradas aquí. Por ahora incluiremos esta noción: si es por encima de cero, “hay motivación”.

Para realizar esta medición que indica el grado en que el docente se encuentra motivado desde el momento que inició sus estudios de carrera, el CM (Componente Motivacional) fue evaluado de acuerdo a la cantidad de veces que se obtuvieron respuestas verbales que coincidieran con los patrones considerados en la primer columna de la siguiente tabla:

Código dentro de Red en Atlas.ti	Datos a prospectar de las entrevistas	Grado de motivación
RE	<p>a) Rechazos en las empresas; poca inserción laboral.</p> <p>Búsqueda: rasgos de resentimiento, culpa, esperanza por ser contratado, deseos de cambiar de rubro. Acciones concretas por cambiar de rubro.</p>	
BA	<p>b) Baja autoestima para enfrentar lo que pudiera encontrarse fuera de la escuela. Búsqueda: baja autoconfianza en habilidades, decepción</p>	
VD	<p>c) Vocación docente adquirida <i>a posteriori</i>;</p> <p>Búsqueda: compromiso con la labor educativa, experiencias movilizadoras con alumnos, comienzo de cursos de pedagogía, comienzo de nueva carrera con elementos pedagógicos: Profesorado de Tecnología, Licenciatura en Tecnología Educativa, compromiso en Olimpiadas de Informática o talleres extras, obtención, recuperación, reciclado o reparación de los equipos a cargo.</p>	<p>-1: Presencia</p> <p>+1: Ausencia</p>
AC	<p>d) Forma en que accede a la complejidad.</p> <p>Búsqueda: <i>rechazo</i> o <i>entusiasmo</i> al cambio y a tecnologías <i>nuevas</i> o <i>distintas</i> (<i>Linux</i> por ejemplo), ausencia o presencia de metodologías de investigación, desapego o incorporación de teorías de tratamiento de la complejidad como Teoría de Sistemas, ausencia o presencia de curiosidad, participación en foros técnicos, proactividad ante problemas recurrentes que podrían ser automatizados, aspecto lúdico ligado a la resolución de problemas.</p>	<p>0: Equilibrio, caso justificado por otras razones</p>
VE	<p>e) Preferencia sincera por las ventajas de trabajar en escuelas:</p> <p>Búsqueda: orgullo por la institución, por el trabajo y sus ventajas, o por el contrario, hastío, excusas, malestar, oprobio, falta de desafíos.</p>	

Para la búsqueda de datos, se opta por revisar una nube de códigos relacionados entre sí, creada mediante el *software* QDA mencionado, Atlas.ti para ir consignando a medida que se relevan los datos y las conexiones con otros conceptos que soporten o descarten las propuestas de explicación.

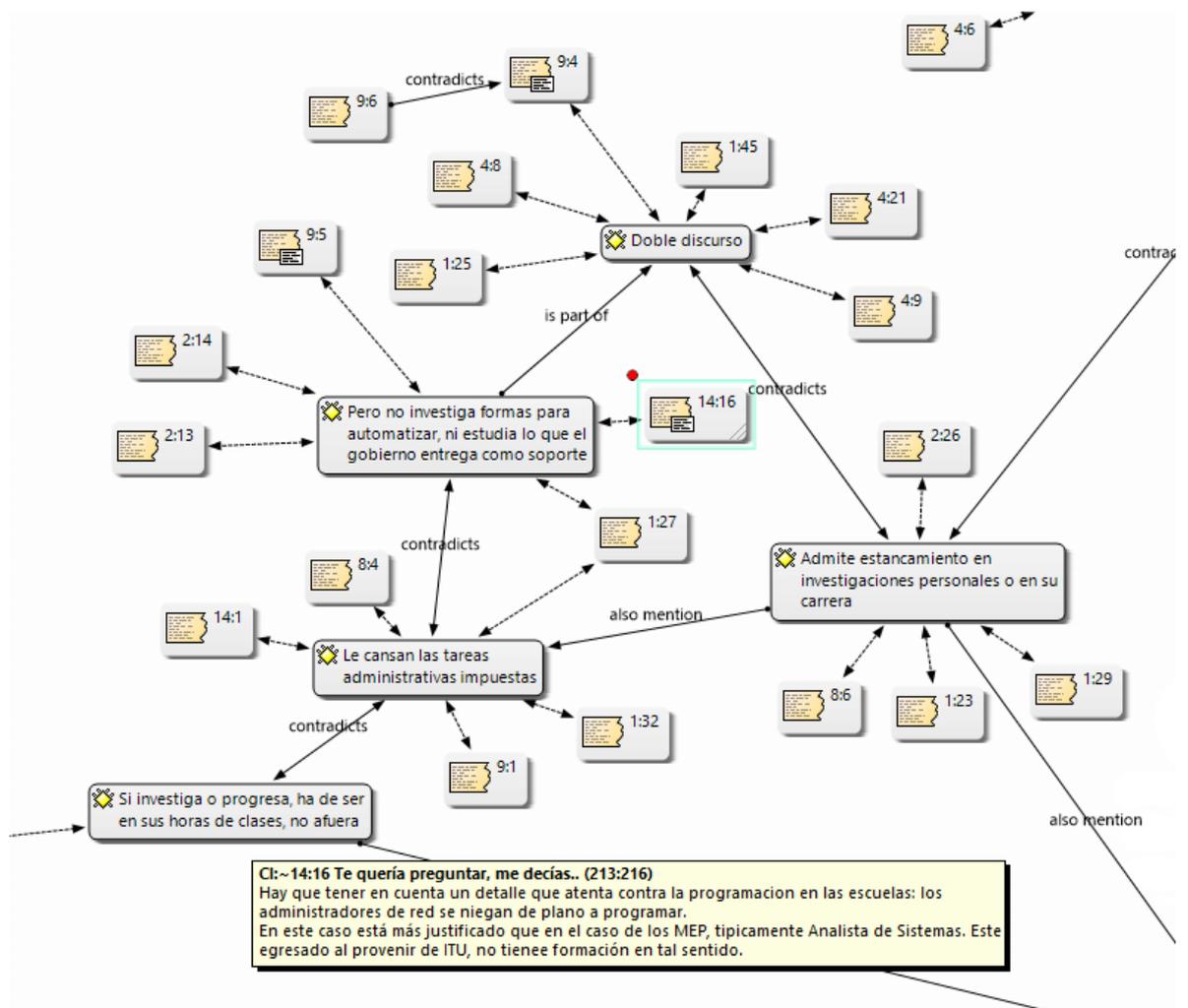


Figura H: Fragmento de la red creada en Atlas.ti durante esta tesina, que muestra conexiones entre códigos y unidades de análisis. Luego se revisarán algunos de los nodos mas prometedoros.

Esta nube de códigos, sus relaciones y sus comentarios pueden encontrarse en el Anexo 2 del CD: Recursos.

Los datos recogidos quedan consignados de la siguiente forma:

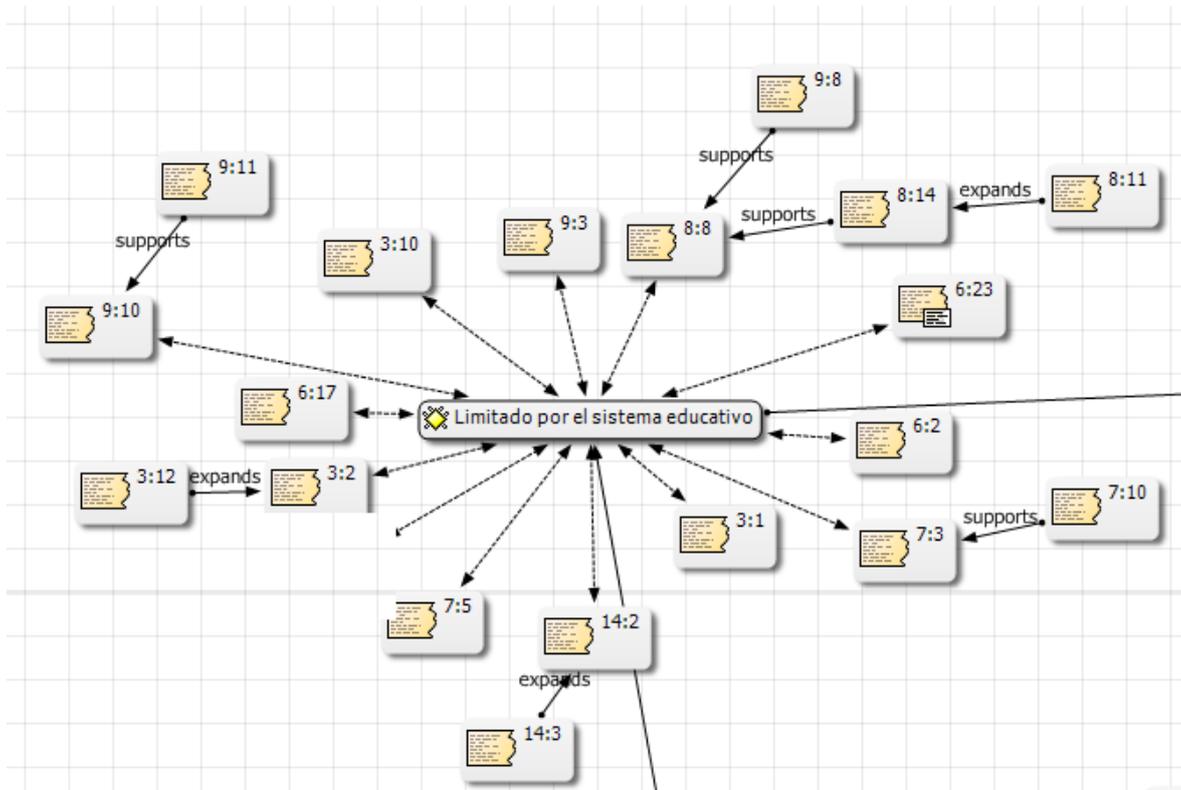
N.º Caso	CM	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		
Orden DP en Atlas.ti (Revisar Anexo 2 en CD)	Componente Motivacional	P1	P8 P9	P4	P2	P6	P17	P12	P13	P18	P11	P49	P14	P5	P15	P10	P3		
A Analista en Sistemas	RE	-1	+1	-1	+1	0								-1			+1	0	
	BA	-1	+1	-1	-1	0								-1			0	-3	
	VD	-1	+1	+1	+1	+1								+1			+1	5	
	AC	-1	-1	0	-1	-1									+1			+1	-2
	VE	+1	+1	+1	+1	+1									+1			+1	7
L Licenciatura en Sistemas	RE				+1	0	-1									-1		-1	
	BA				-1	0	0									-1		-2	
	VD				+1	+1	+1									0		3	
	AC				-1	-1	-1									0		-3	
	VE				+1	+1	+1									+1		4	
I Ingeniero	RE							0	+1	+1			+1		-1			2	
	BA							0	+1	+1			+1		0			3	
	VD							-1	+1	0			+1		+1			2	
	AC							+1	+1	+1			0		+1			4	
	VE							-1	+1	+1				+1		+1		3	
O <sub>T</sub> Otros: Técnico en Redes	RE								+1			+1	+1					3	
	BA								+1			+1	+1					3	
	VD								+1			+1	+1					3	
	AC								+1			+1	0					2	
	VE								+1			0	+1					2	
O <sub>D</sub> Otros Diseñador Industrial	RE										+1							1	
	BA										+1							1	
	VD										-1							-1	
	AC										0							0	
	VE										+1							1	
O <sub>P</sub> Otros: Programador	RE				+1													1	
	BA				-1													-1	
	VD				+1													1	
	AC				-1													-1	
	VE				+1													1	
	RE					0												0	

O <sub>TU</sub> Otros: Técnico Universitario	BA					0													0
	VD					+1													1
	AC					-1													-1
	VE					+1													1
P Profesor en Informática	RE											+1	-1						0
	BA											+1	-1						0
	VD											+1	+1						2
	AC											0	+1						1
	VE											+1	+1						2
P <sub>T</sub> Profesor en Tecnología	RE																	-1	-1
	BA																	0	0
	VD																	+1	1
	AC																	+1	1
	VE																	+1	1
P <sub>F</sub> Profesor en Filosofía	RE																	-1	-1
	BA																	-1	-1
	VD																	0	0
	AC																	0	0
	VE																	+1	1

Código dentro de Red Atlas.ti	Promedio	Comentario
RE	0,8571428571	No incluye profesores, solo docentes habilitados por sus títulos.
BA	0,1428571429	
VD	2,00	
AC	- 0,1428571429	
VE	2,7142857143	

Si se hace un promedio final de estas cifras, obtenemos que el CM es de 1,11428571428.

Se mencionó que positivo es “motivado”. ¿Es mucho? ¿Es poco? Una comparación con otras regiones podría arrojar más luz sobre el tema. Por ahora, sabemos *cuales variables* bajan o suben esta cifra. Por ejemplo, la cifra que corresponde a VE (Ventaja de trabajar en la Escuela) es alta: los beneficios compensan la BA (Baja Autoestima) descrita por ellos de acuerdo a diversas limitaciones, falencias, carencias de recursos, vicios del sistema educativo, arbitrariedades cometidas por directivos y cuerpos contralores, y las injustas (o no), tareas desmotivantes de los que son objeto.



*Ilustración I: Malestar, oprobio, hastío, excusas, sensación general del trabajo*

Si a pesar de tantas ingratitudes, los docentes se quedan en la escuela y no se emplean en las empresas, entonces cabe poner el foco en BA. Cuando se observan las cifras, este valor se halla correlacionado inversamente a una potencial baja de capacidad de abrazar casos complejos (AC), o al menos de deconstruir sus partes y rearmarlas<sup>45</sup>:

$$BA = 0,1428571429; AC = -0,1428571429.$$

Si se restringiera tanto el análisis a nivel cuantitativo, como las primeras inmersiones cualitativas, la conclusión sería lapidaria, máxime si se toma en cuenta varios testimonios en común:

45 Las cifras inversas que muestran estas variables son llamativas por la exactitud de sus decimales. Para explorar la casualidad o, por el contrario, posibles patrones, se adjunta a este documento un archivo *Hipótesis 1.2.ods* en formato hoja de cálculo de *LibreOffice*.

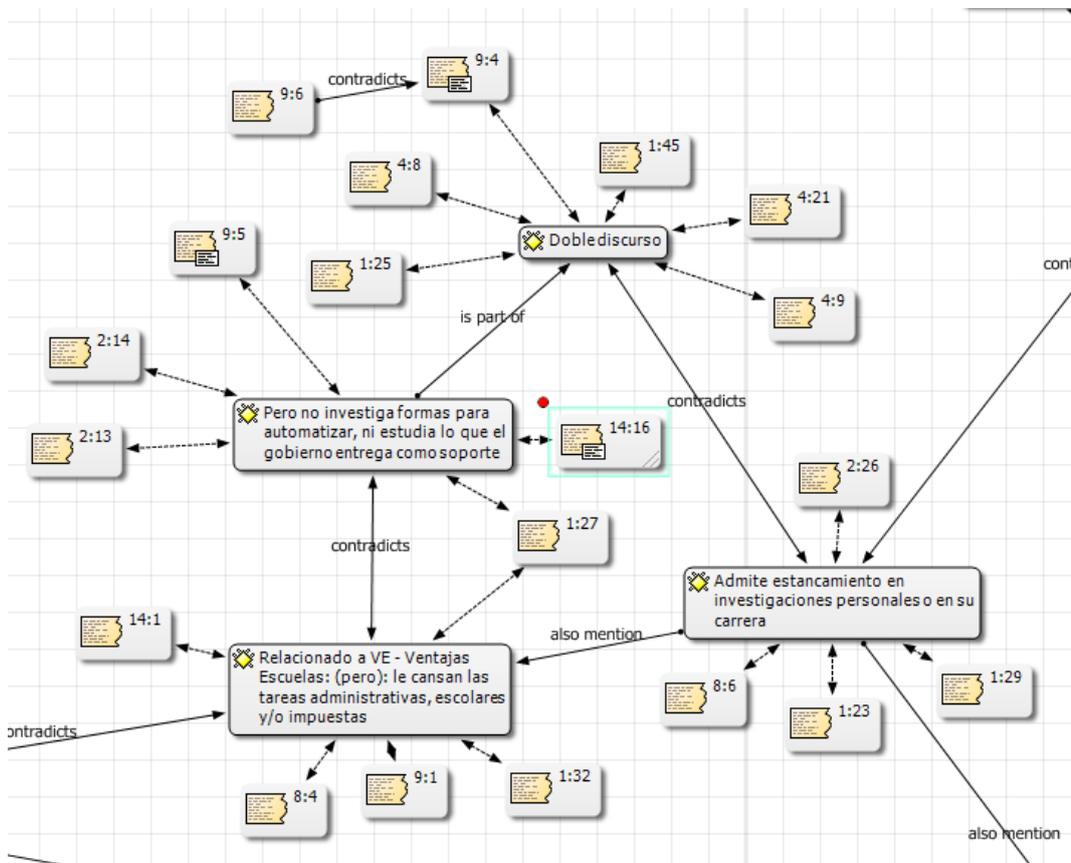


Ilustración J: Baja Autoestima y Acceso a Complejidad, fragmento de red en Atlas.ti

De esta manera AC se convierte en el eje de esta hipótesis. Con la responsabilidad de no caer en subjetividades, más adelante, en la conclusión, se mostrará la construcción de una herramienta capaz de medir la complejidad técnica en la escuela con un mayor detalle. Sin embargo, en este punto, el AC cuantitativo puede terminar de redondearse en forma cualitativa, consultando por formas de automatización en la escuela.

## Geeks and repetitive tasks

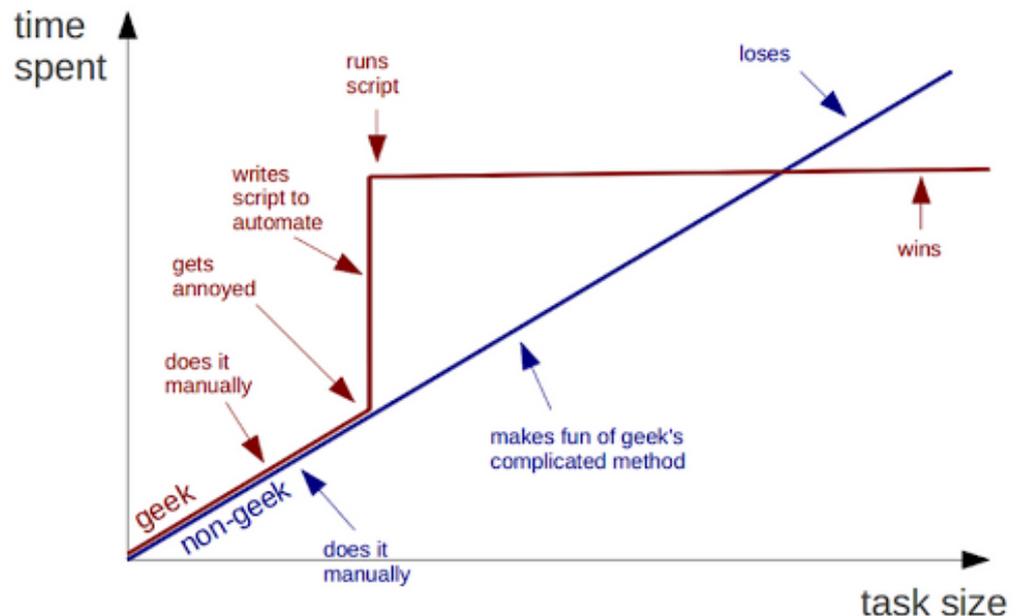


Ilustración K: automatizando una tarea repetitiva ("meme" de procedencia desconocida).

Para entender esta idea, hace falta poner en contraste el principio DRY "Don't Repeat Yourself", que se aplica en las empresas en gran medida por la influencia de la "regla de practicidad" publicada en *La Catedral y el Bazar* de Eric Raymond (2000): cuando una tarea se vuelve rutinaria, los informáticos la *automatizan*<sup>46</sup>. El ejemplo clásico de *no automatizar* en las escuelas es cargar los datos múltiples veces, en varias hojas de cálculo o paginas web del gobierno, en lugar de concentrarlos en un solo lugar.

Precisamente los informáticos de las escuelas que fueron entrevistados se quejan del trabajo administrativo. ¿Qué han hecho para resolverlo? Hasta donde se pudo constatar en el presente, las soluciones que propusieron se han basado

46 La cita original es "Every good work of software starts by scratching a developer's personal *itch*", y forma parte de otros conceptos que se revisan en el Apéndice A.4.3. Protointerfaces resistentes *ex professo*.

en el tratamiento de ofimática avanzada: empleo en profundidad de *Excel*, *Access* y *Google Drive*. Es decir, la complejidad esperable de un operador experto en PC, o de un administrativo sobresaliente.

Si se continúa este orden de ideas, no se halla, al menos en las 18 entrevistas realizadas, casos de creación de programas que ayuden a la planta docente, permeabilidad con padres vía Internet, uso de motores de bases de datos, intentos de acceder la API de Anses o de Dirección General de Escuelas, el *scrapping* de datos ya existente, la programación de formularios bajo *Intranet* o la aplicación de alguna de las soluciones aprendidas durante la carrera.

Es cierto también que los docentes de informática y encargados de laboratorio han encontrado resistencias en directivos, cambios de escalas año a año o tiempos acotados en sus horarios, pero ni siquiera se han identificado casos donde hubieran instalado un *software* libre de gestión de escuelas como ALBA, Fedena o MicroISIS (para Bibliotecas). Menos aún casos donde hubieran adaptado, corregido o colaborado con estos proyectos mediante alguna herramienta de gestión comunitaria de código como SVN o GIT.

De todos los casos examinados hubo uno solo de instalación de plataformas de *e-learning* con *Moodle*, merced posiblemente a la insistencia de los coordinadores del proyecto Conectar Igualdad; también hubo una prueba detectada para configurar ALBA. No ha habido casos de desarrollo de proyectos de construcción de *software* para automatizar la gestión escolar, ni impulsada por la escuela, ni emprendida para rédito personal, ni en forma colaborativa y pública.

De hecho, la participación en foros relacionados con estos temas es nula. Si se hace una revisión en los foros de educ.ar, program.ar y gleducar se puede reconocer a docentes colaborando y participando de estos temas. Pero el número de consultas y respuestas por mes es tan bajo para la población de docentes

considerados a nivel país, que los casos de profesionales inquietos y comprometidos con la enseñanza de tecnología parecen más casos extraordinarios que comunes. Por esta razón las entrevistas obtenidas poseen el grado de generalidad suficiente como para sostener que hay un problema derivado de la propuesta original: "de contar con el suficiente razonamiento lógico-matemático, o al menos una suficiente fuerza de voluntad, los alumnos de las carreras tecnológicas tenderían a finalizar sus estudios, incluso si han perdido toda la pasión que los ha motivado a emprenderla." [H<sub>1,2</sub>].

La hipótesis planteada originalmente se ve ampliada al incluir la noción de que los estudiantes de las carreras confían que el solo hecho de contar con un título asociado al campo, les permitirá trabajar en empresas, pese a notar mucha dificultad durante sus años de cursado. Esta esperanza guía e impulsa sus estudios. Varios entrevistados aportan este dato: la "moda" de la informática auguraría éxitos para conseguir empleo. Incluso en el caso de haber una falta de confirmación batesoniana de su *self*, es decir, la confirmación de que no estarán a la altura de la imagen proyectada al inicio de sus estudios, varios de ellos ya cuentan con el plan B (las escuelas) antes de egresar. Otros se cambian al profesorado, pero no por una concientización respecto de sus capacidades. Éstos lo hacen por vocación, y se percibe porque las materias de pedagogía exigen al menos la misma dificultad que las materias técnicas: por su carga teórica son incluso más difíciles de abarcar para aquellos con pensamiento determinista / mecánico, que son los ingresantes clásicos a las carreras de sistemas. Esta es una sana elección que les aportará herramientas pedagógicas para acometer con mayor dignidad y precisión su trabajo aúlico.

De esta manera se presenta la escuela como plan B, como una "buena decisión", por cuanto la vocación docente adquirida (VD) es superior a 1

("presencia" igual a 2) y las ventajas consideradas de trabajar en las escuelas es muy alta (VE: 2,7).

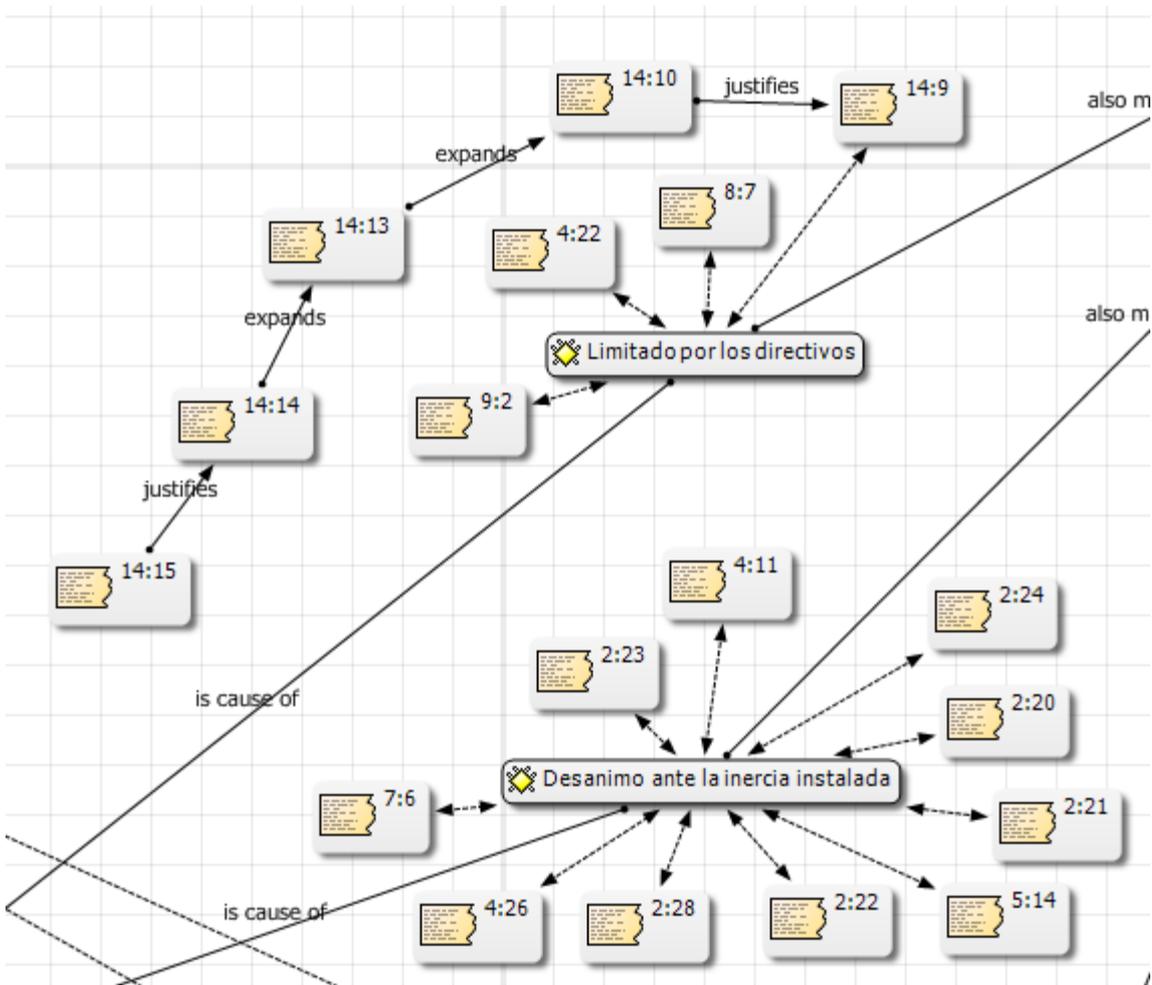


Ilustración L: Sensaciones de limitación, desanimo.

La zona de conflicto se pone de manifiesto cuando se retoman los indicadores propuestos: 1) *decepción* durante el ejercicio de la profesión; 2) el ejercicio de *otra actividad*; y 3) la *docencia exclusiva en tecnología* (pues rehuye ejercer en empresas).

Respecto del primer indicador existen múltiples testimonios de sentimientos de decepción. Se propone, por tanto, una "acción terapéutica": evaluar estos aspectos durante los exámenes psicofísicos. El AC, si bien no es un condicionante para trabajar en escuelas, debe ser traído al plano consciente junto a un interlocutor psicólogo, puesto en relevancia y tratado de tal forma que no impacte ni en la educación tecnológica de los educandos, ni en la calidad de las soluciones a brindar en las escuelas. En tal sentido hay una suerte de contraposición marcada entre los valores altos de VE y de VD, con relación a la multitud de quejas de los informáticos sobre el sistema educativo, y a la pobreza de acciones para resolverlo comenzando *in situ*, una suerte de bucle de Watzlawick que se tratará en la conclusión de la presente obra.

Esta carencia de acciones puede confirmarse en el segundo indicador ("ejercicio de otra actividad") y en el tercero ("docencia exclusiva en tecnología"), ya que las muestras obtenidas para revisar este indicador exhiben ausencias marcadas en investigación, aut Capacitación y actividades profesionales extras al trabajo diario docente. Esto provoca una falta de entrenamiento en la gestión de la complejidad, un abanico limitado de soluciones de automatización para las escuelas y una recarga en el aparato público.

Esta carencia de acciones, además, requiere del ofrecimiento de una imagen completa del fenómeno, puesto que el AC no es único problema. El indicador 1 ("decepción durante el ejercicio de la profesión") se da precisamente *en la escuela y con la escuela*, lo cual se podría comprobar si contrataran a un desconocido "Bill Gates": sus ideas serían desechadas de plano, se le impondrían tareas administrativas comunes y, en forma explícita y contraviniendo sus funciones de *facilitador*, quedaría prontamente relegado a realizar tareas de *data entry*.

Obviar este aspecto equivale a cercenar la mejor forma que tienen los encargados de seguir aprendiendo: aplicar sobre el terreno lo aprendido en la

universidad. La acción del directivo es clave aquí, y su visión limitada de la tecnología implica un grave desaprovechamiento de un valioso recurso. Un ejemplo simple de esto son los onerosos servicios de marcado de horario del personal por huella digital que se contratan en las escuelas, cuando estos pueden reemplazarse fácilmente por una vieja PC, una rudimentaria base de datos, una cámara web y un pequeño algoritmo que dispare una foto al momento del marcado.

En este sentido hay múltiples referencias respecto del nulo acompañamiento que hacen los directivos respecto de los cambios que puedan aportar los encargados de laboratorio. Solo por traer uno de los ejemplos recogidos, en las entrevistas a tres encargados de la Escuela Normal 9-002 "Tomás Godoy Cruz", una vez que los preceptores contaron todos con sus *netbooks*, se pudo relevar la imposibilidad de introducir un sistema para integrar las faltas escolares de los alumnos. El personal docente normalmente se niega a efectuar cualquier tarea que implique un gasto extra en su consumo cortical de energía, si esta tarea puede ser resuelta *manualmente* desde el Laboratorio de Informática por un informático que pase los datos a Excel.

Si bien el problema del cambio ("cambio" como objeto de estudio) es normal en las empresas, su aplicación en el caso de las escuelas puede llegar a presentar niveles muy altos de conflicto. Allí la capacidad de gestión del directivo es el eje a considerar. La propuesta en este sentido es realizar jornadas para directivos donde se les enseñe *cómo aprovechar al encargado de laboratorio*, cuáles son sus usos, límites, mostrar ideas y casos de éxito<sup>47</sup>. Fomentar entre los encargados el

47 Un interesante avance en este sentido se encuentra en la página de Program.ar, con un curso que se presenta como "Gestión y liderazgo de proyectos creativos que integren las Ciencias de la Computación en las instituciones escolares".

Anónimo (2016). *Curso para Directores y Supervisores*. Recuperado de <http://program.ar/formacion-docente/>, [fecha de consulta: 29 de agosto de 2016]

equivalente a "Ferias de Tecnología", *workshops* y congresos donde puedan compartir soluciones.

El otro aspecto al que los entrevistados se ven expuestos es la distracción constante. Si bien es natural que los encargados de laboratorio se conviertan en referentes tecnológicos, se subestima explícitamente la concentración que se requiere para compartir selectivamente los datos de la escuela con el exterior, protegerlos y facilitar sus accesos. Ello equivale a contratar un matemático capaz de realizar cálculos complejos, para ocuparlo resolviendo cálculos elementales al alcance de una calculadora de bolsillo o "con papel y lápiz".

En efecto, un tema del que no se habla en las escuelas es el de la inconveniencia de eximir de tareas administrativas al personal de laboratorio, como no corresponda estas a tareas relacionadas con programas de alfabetización digital. De hecho, en ninguna de las funciones de los MEP (Maestro de Enseñanza Práctica) de Informática, ATP (Auxiliar de Trabajos Prácticos) de Informática, AIE (Auxiliar de Informática Educativa) se incluye tales obligaciones. Sin embargo, las tareas administrativas son demandadas como prioridad absoluta por sobre las funciones del puesto, y exigidas perentoriamente por cualquier miembro del personal docente.

En virtud de lo expuesto se debe considerar que el CM debe ser alto entre los informáticos, especialmente elevado si trabajan en escuelas. No importa si trabajan en escuelas, en empresas o si son *freelancers*, el tipo de profesión al que se enfrentan requiere de estar preparado para la ingratitud constante —ya se ha visto que no se les menciona este tema en los Pre Universitarios. El único CM con el que se cuenta es la motivación intrínseca al comenzar su carrera. Y si trabaja en las escuelas, si bien aumentará su VD y su VE (de otra forma renunciaría), este CM tenderá a disminuir. Si además se considera la figura de referente informático, se debería incluir la noción de que el CM se erosiona conforme aumenta la

sobrecarga de trabajo administrativo, las consultas por problemas técnicos personales y otros factores como impaciencia y exigencia de resultados inmediatos respecto de situaciones técnicas que deberían ser simples, y que no siempre lo son.

Tras un rudimentario rastreo longitudinal de la estructura organizacional del área de Laboratorio de Informática, esta percepción de "estado de urgencia constante" pudo relevarse a lo largo de quince años en el laboratorio de la mencionada Escuela Normal 9-002 "Tomás Godoy Cruz". En este transcurso siete encargados de laboratorio no conectados temporalmente entre sí, merced a la cultura institucional del establecimiento, han de prestado servicios a toda la planta docente y reemplazaron en sus tareas administrativas a los usuarios si éstos no podían o presentaban resistencias al uso de nuevos dispositivos tecnológicos. A pesar de que los encargados se deben a los alumnos y a las normativas emanadas por Dirección General de Escuelas y sus programas específicos (Conectar Igualdad o Uno a Uno). Dichas tareas son exigidas perentoria y permanentemente, lo cual se puede constatar a través de testimonios registrados en las entrevistas. En ellas queda corroborado que el caso de la Escuela Normal no es aislado, sino frecuente en la mayoría de las escuelas relevadas. Entre los muchos ejemplos posibles, se destacan dos de ellos.

En el primer testimonio, el encargado de laboratorio traza los límites de su función, y aunque no está obligado, colabora de una forma más provechosa tanto para él como para la escuela (colabora, acompaña, aprende, "hace los backups" en el proceso), y dice lo siguiente:

B: —El trabajo acá, que estoy haciendo en la escuela, me gusta mucho. Está muy bueno, pero porque yo siento que estoy haciendo el trabajo que me corresponde. En otras escuelas te decía lo que pasa... llega la Directora y dice "yo necesito que alguien haga los analíticos". Hay

muchos casos de administradores de red que están haciendo analíticos, que es llenar... te metés a un programa y llenás las notas de los alumnos. Acá lo hace una preceptora; yo le ayudo, o sea, le ayudo a hacer el *backup*. Si falla la computadora, en eso; pero ella lo carga.

A: —Claro, soporte técnico, pero ella es quien carga la información.

B: —Eso, pero es porque bueno... También yo creo que va mucho con qué actitud tenés, si vos llegás a la escuela y tenés una actitud que viene la directora y te dice: "Mirá vos hacé esto". Y vos decís: "bueno". Y te quedás callado, te vas a quedar así.

(cf. Anexo 1: Entrevista P14).

No ocurre lo mismo en el segundo caso, donde una vez sobrepasadas sus funciones, no logra escapar a las mismas, y entra en una meseta de habilidades, como se puede apreciar en la siguiente entrevista:

19. *¿Cuáles son las últimas innovaciones que has realizado en tus máquinas o dispositivos tecnológicos?*

A: —¿Has hecho alguna innovación en tu *netbook*? ¿En tu compu? Alguna cosa que... no sé... ¿Alguna cosa que hayas armado y te haya hecho sentir orgulloso?

B: —No, no, no, no. No.

A: —¿No te ha pasado?

B: — No, no me ha pasado, porque como te digo, me he estancado como los mejores —risas.

A: —Está bien, quizás es una sensación propia.

B: —Sí, por ejemplo, por ahí me decía, me decía...: "Ahora voy a hacer una base de datos". Por ejemplo, decía con el tema de las *netbooks*, tener una planilla en Google de una planilla de cálculo, decía, a ver, "bueno... a ver qué manera". Bueno, veía las relaciones, alumnos, *netbooks*, reasignaciones. Cosa de estar todo en una base de datos, aunque sea en una base de datos común en Access. ¡Mejor chiquita! Donde, bueno, eh... tenga el control y diga "Bueno, a ver...", "Se hace un reclamo al servicio técnico, la *netbook*" y lo tenga anotado, y digo "A ver. Decime tu apellido", "buscá a fulanito", Vero, "¿a dónde está...?", "rastreámelos y que esté todo. Todo vinculado. Y me traigás...", "Vos. Cargamos tu máquina con un problema técnico, tu número de referencia es "tanto", ¿me entendés?".

A: —¿Te sentís motivado para emprenderlo? ¿Qué es lo que hace que después digas "está bueno como idea" pero [después] no lo seguís?

B: —El tema... —indica tras una pausa prolongada.

A: —¿Parezco un psicólogo?

B: —Sí, nos estamos yendo para otro lado —expresa evidenciando cierta incomodidad—; no, el tema de que por ahí... o sea... este... —hace pausa larga— Sí... Es mucha...

A: —¿Te preocupa terminarlo y que te critiquen?

B: —No, terminarlo, no.

A: —¿Puede ser que lo hacés y después te dicen: "tiene un error"?

B: —Nooo. El tema de que... —realiza una pausa—; yo como que veo... como que... entré en el mundo de los informáticos cómodos —entrevistador y entrevistado se ríen.

A: —¡Eso es honestidad! Está bien, entonces digamos, no vislumbrás que haya una motivación importante para seguirlo digamos [...].

A: —Ves... ves... Yo estoy planificando para hacer..., quería hacer... estaba empezando... para hacer... motivarme para hacer... Bueno, a ver alumnos, las reasignaciones, estaba haciendo hoy día, estaba embolado y lo estaba haciendo..., pero ¿el tema cuál es? Para hacer, viste, que sé yo, porque estaba embolado hoy día y ella me estaba hablando de.... no me acuerdo de qué estábamos hablando. Entonces, bueno, estaba por hacer algo, ¿viste? Tenés ganas de hacerlo, entonces, ¿qué pasa? Soy como un informático acomodado... cómodo.

A: —Digamos, tu motor sería el aburrimiento, por así decirlo. ¿Tenés que estar "hiperaburrido" para ponerte a hacerlo? —ambos ríen.

B: —No... no sé si por aburrido, lo que pasa es que bueno... Yo te digo mi rutina: yo llego, yo estoy, salgo a las siete de la tarde, me tomo el... seis de la tarde, me tomo el colectivo y llego a las siete y media. Ahí, a las ocho, ahí a las ocho de la tarde me voy al gimnasio porque me gusta irme a hacer mi rutina de actividad física. Vuelvo, ceno y la cabeza por ahí me da un puñet[azos] —no se entiende— si todos me han hinchado que el tema de cargar denuncias, los chicos... Los chicos te agotan, te agotan —afirma estas últimas palabras.

B: —Te agotan. Se acercan... los chicos: hay un problema. Los alumnos de la escuela en general, y a lo mejor del Normal más, son más

prepotentes. Entonces, que pasa, te caen todo el tiempo, te caen por las *netbooks*, y el problema, ¿sabes cuál [es el] más grande? Que nosotros somos los tontos que nos dejamos hacer, es que nosotros nos dejamos que ellos nos falten el respeto y nos pasen por arriba.

A: —Está bien, digamos, [la] excesiva operatividad que te provocan los pibes.

B: —Pero, ¿qué pasa? Salís cansado negro. Yo salgo cansado. Yo me levanto como todos, más a las seis y media de la mañana me vengo de allá hasta acá [nota: vive en San Martín]. Acá a la escuela, una de las escuelas. Almuerzo en el centro, voy de una escuela a la otra, salgo, me voy a mi casa, voy a hacer actividad física, porque ¿te imaginas? Estás siempre sentado. De que de ahí... de ahí que llegue a mi casa, que cene que tomo algo y la cabeza no me da. Ya estoy... como que he entrado de decir "basta".

Más adelante agrega:

B: —Te vuelvo a repetir, o sea, esto es por una cuestión de quedado yo. Yo no tengo drama de decir que es "por quedado", ¿sí? Eh, pero el tema es por ejemplo que a mí me gustaría, por ejemplo, no sé a ver... eh... explorar o por ahí por ejemplo investigar un tema de Android, cosas así, ¿viste? Cualquier cosa. Por ahí yo que salgo a las 7 de la tarde que llego a mi casa. ¿Qué voy a hacer...? Voy a gimnasia... Voy allá. ¿Viste? Por ahí es medio como que... pero... Este... No, en el trabajo no. Nunca me ha dado...

A: —¿Y si te obligaran? En el sentido... No sé. Viene un día el director y te dice, "Ahora queremos...". No sé, [un programa en el] que los alumnos puedan ver las notas.

B: —¡Ay, no! —contento. Eso... Un día se me ocurrió a mí el año pasado. Digo... "Bueno. Vamos a hacer primero que carguemos en un sitio *web*. Ponele... Las notas de los chicos para que los chicos puedan consultar. Hagamos una aplicación..., un [paquete] APK para Android que directamente los chicos, desde el celular, consulten nada más, que consulten desde el servidor y ver cómo está su situación escolar. ¿Me entendés?

A: —Estaría bueno.

B: —Eso se me había ocurrido. ¡En serio! ¡Te lo juro! ¡En serio!

A: —Pero ¿lo alcanzaste a proponer? ¿Le dijiste?

B, realiza un movimiento de negación con la cabeza: —¿Sabés qué pasa? Es por quedado. A mí no me interesa si proponerlo o no proponerlo. Yo lo puedo hacer.

A: —¿Y si te cambiaran todas las tareas que estás haciendo ahora porque te digan: "Sí, está buena la idea. Dedicate nada más que a eso."?

B: —¡Sí! Lo hago de una.

A: —Entonces ahí tenés una recompensa. Digamos.

B: — Sí. El problema, ¿sabés cuál es?

A: —¿Qué vos trabajás por diversión básicamente?

B: —Es que yo... Me gusta... me gusta en serio. ¿Sabés cuál es el problema que tengo yo? Y me da vergüenza decirlo por ahí, es que hace tanto tiempo que no veo Java que me da vergüenza cómo encarar, cómo empezar. Todo ese proceso de investigación, todo el tema de documentación. Todo... Por ahí. Si vos me das... y yo no me quiero hacer el... Haber... Si vos me decís, por ejemplo: "Vamos a ver... este manual, este tutorial, este video de cómo se programa por ahí el Android, que yo he estado viendo, que he estado trabajando, [que] tengo en mi casa, en mí *notebook*, el emulador... Hacía como dos años atrás y lo tenía en Lin... en Ubuntu lo tenía al emulador...

A: —¡Ah! lo tenías instalado ahí y todo.

B: —Sí, y como andaba... ¿Qué sé yo...? Vos decís si yo me hubiese dedicado a eso... Bueno... No tengo drama de hacerlo. El tema es cómo encarar lo que es lo básico. Yo, por ejemplo, para hacer un sistema, como vos... No podés saber todo... A ver... Un código fuente... un lenguaje de programación lo podés manejar bien, pero el tema es que no sé todavía, o no me doy idea de cómo encararlo. Por ejemplo, para que sirva, que sea útil el sistema. ¿Me entendés? El emulador de Android yo lo he instalado 3 veces y las 3 veces lo he dejado (cf. Anexo 1: Entrevista P1).

De esta manera, el estudio del CM, contemplado en su indicador 1 ("decepción durante el ejercicio de la profesión") se encuentra ligado a sus expectativas originales. Para medirlo en profundidad requiere por lo tanto una exploración longitudinal en sí misma para terminar de cerrarlo.

Sin embargo, se sabe que hubo una motivación para ingresar, y que luego esta se perdió. La pérdida del CM, en cuanto objeto de estudio, requiere un

abordaje desde varias perspectivas. Ello se patentiza en varias de las preguntas de las entrevistas las cuales apuntan a ello, y sus respuestas más emblemáticas pueden encontrarse en la red de nodos que acompañan al proyecto en Atlas.ti —cfr. Anexo 2 del CD: Recursos, conectados al margen derecho de cada una de las entrevistas.

Entre las preguntas que se hicieron expresamente para detectar el CM *antes*, *durante* y *después*, se hallan preguntas tales como: "¿Qué te motivó a estudiar tu carrera?", "¿Tenías acceso a una computadora durante tu infancia o adolescencia?", "¿Tenías consola de juegos?", "¿Qué cambios de perspectivas realizaste desde tus comienzos hasta la actualidad?", "¿Qué docentes te marcaron más en tu comprensión de la tecnología y el manejo de sistemas informáticos y computacionales en tu carrera y por qué?" y "¿Qué te hubiera gustado aprender que no te enseñaron?".

La punta del hilo puede establecerse en el factor detectado según el cual los encargados de laboratorio y los docentes, si bien son muy competentes en sistemas *Windows* y en ofimática instalada sobre PC, estas piezas de *software* tienden a ser las únicas que usan para resolver problemas. Volviendo al ejemplo de resolver —sin costo monetario— el problema de marcado de horario del personal, no se pudo constatar, por ejemplo, el armado de ningún *Raspberry* o *Arduino* que lo resolviera. La construcción de semejante dispositivo escapa a lo que se pretende de ellos en una escuela. Pero este ingenio puede ser montado también en una *Netbook* que apenas funcione: el algoritmo planteado hipotéticamente solo necesita el siguiente bucle, a saber:

- 1) solicitar un D.N.I. por pantalla;
- 2) hacer "algo" que capture una foto desde la cámara<sup>48</sup>; y

48 Por ejemplo en Linux Huayra, el siguiente comando toma una foto desde la cámara

3) guardar D.N.I. y foto en una base de datos.

Pero, ¿y si el directivo forma parte del problema "de no poder resolver el problema"? Estas tres instrucciones concisas y claras son un ejemplo de las órdenes que se emiten en las gerencias: «¿se puede resolver "x" problema mediante algún dispositivo tecnológico disponible entre los recursos existentes?», «¿se puede integrar "x" datos en un programa que nos resuelva "y" problema?», etc. El *cómo* lo debe resolver el informático.

En tal sentido distinguir el hecho que el directivo sea una figura escolar y que el gerente sea una figura empresarial es inadmisibile, puesto que ambos experimentan diversos problemas de gestión, ambos tienen que ser capaces de plantear, en líneas generales, aquellos problemas que más los aquejan y colaborar inicialmente en el razonamiento que lo resolvería proveyendo los requerimientos específicos que se precisa resolver, y que el personal de informática ha de efectuar.

La formación de los directivos no los suele preparar para saber pedir o requerir acciones en las distintas áreas de especificidad de una institución o empresa. Por lo tal motivo no suelen ser capaces de plantear estos "requerimientos de producto"; de acuerdo a la jerga *Scrum* de producción de *software*. No obstante sucede que tampoco ellos necesitan interesarse en detalles en cuanto a la forma de la solución, de modo tal que baste con completar tanto manual como automáticamente las producciones realizadas.

El informático, por su lado, también forma parte de la subestimación que se hace de su cargo. Basta con que "acostumbre" a brindar soluciones de este tipo a los docentes que lo rodean, para dejar de ser relegado al papel de un técnico "que

```
fswebcam -d /dev/video0 -r 800x600 dni.jpg --set brightness=100%
```

Este comando puede ser "llamado" desde cualquier lenguaje de programación.

Sergio A. Alonso - 170 - Resistencia al cambio en torno a las NTIC

transcribe planillas". Es él entonces quien debe romper el ciclo, con aplicaciones "de fondo" (*backend*), en lugar de aplicativos ofimáticos (*frontend*), tales que solucionen precisamente, problemas de fondo<sup>49</sup>. ¿Por qué entonces el informático no lo hace? ¿Por qué no emerge?

Si se parte de estos interrogantes relativos a la *Grounded Theory* planteada por Barney Glaser y Anselm Strauss, cabe plantear una hipótesis del estilo de H<sub>1.3</sub>. Dicha hipótesis propone que la equivocación en la elección de carrera podría tener una conexión a partir del momento en que el aspirante cree que llevar su carrera en sistemas será tan fácil como manejar los programas típicos de PC que usaba antes de ingresar, aquellos programas básicos de *frontend* aprendidos como única opción en su propio nivel Medio, donde no ha tenido programación ni algoritmos (¿acaso el inicio del problema?). Así el cursado de sus carreras superiores plantearía diversas incomodidades temporales, susceptibles de ser superadas tras interponer cierta cuota de voluntad e insistencia. Tras su resolución cabría la posibilidad de volver a esta esfera de cualidades entrópicas, en los siguientes sentidos: a) que la ofimática por si sola no soluciona el problema de gestionar *grandes* cantidades de datos; b) que estos datos tienden a desordenarse con el tiempo; y, finalmente, c) que la urgencia resultante condiciona a cortar el círculo. La contrastación de tales aseveraciones se enfrenta a la misma resistencia al cambio. Un informático acostumbrado a tales modalidades no solo no usa herramientas *backend*, ni siquiera es capaz de cambiar su propia interfase *frontend*. Ello es una consecuencia y, a la vez, una causa —por su índole recursiva—, de no poder usar *Linux* como sistema operativo en temas cotidianos,

49 Una aclaración respecto de los términos *frontend* y *backend*. Estos se emplean aquí en su acepción anglosajona, considerados dentro del *argot* que sitúa al primero en la capa 7 del modelo OSI (aplicaciones de usuario); en tanto que el segundo se sitúa en las capas inferiores (sistema operativo, base de datos, *hardware*, *firmware*, protocolos, librerías y demás "plomería" que sustenta a todo aquello que encuentra el usuario).

apoyarse en él para tareas específicas (por ejemplo, servidor de archivos) y a veces, ni siquiera poder usar Libre / Openoffice como *suite* de oficina.

La conjetura H<sub>1.3</sub> plantea que "La exposición inicial a *interfases* enriquecidas sería directamente proporcional a la resistencia al cambio de *frontend*, e inversamente proporcional al aprovechamiento del *backend* subyacente"<sup>50</sup> se podría simbolizar del siguiente modo:  $EIE = RCF/ABS$ , donde EIE representa la exposición a *interfases* enriquecidas, RCF simboliza la resistencia al cambio de *frontend* y ABS remite al aprovechamiento del *backend* subyacente.

La dimensión a considerar para este estudio se encuentra en la capacidad de lograr el estadio final de la línea roja mostrada en la Ilustración H —cfr. supra.

De acuerdo a las experiencias recogidas en empresas e instituciones donde se aplica metodología DRY (o "no hacer dos veces lo mismo"), hay una correlación entre el CM y el ABS, es decir que logrado un cierto grado de automatización sobre las tareas rutinarias escolares, se puede arribar al ideal del cuadrante II planteado por Stephen R. Covey (2003, pp. 91), donde el tiempo se emplea mayormente para capacitarse, planificar y mejorar la calidad del trabajo, en contraste con el cuadrante I, donde el tiempo es afectado continuamente por urgencias. Estas representaciones guardan cierta analogía con los términos *proactivo* y *reactivo*, puesto que siendo *proactivo* se reduce la cantidad de urgencia, en relación con el término *reactivo* porque atiende solo las urgencias y

50 La distinción que se hace sobre las *interfases* enriquecidas, en caso de limitarse éstas a las comerciales, se refiere a que las primeras serían más amables y más elaboradas que aquellas con licencias libres, o en todo caso, que las CLI o "*Command Line interfase*". Respecto de las *interfases* libres, el concepto se encuentra desarrollado en el apéndice A.4.3. "Protointerfases resistentes *ex professo*", en tanto que las CLI se encuentran desarrolladas en el apéndice B.5. "La CLI como protolenguaje de programación".

presiones del momento (estatificación defensiva frente a las demandas del momento, una suerte de inercia dinámica de supervivencia perentoria).

En cuanto a la población muestreada solo se indagó y midió el trabajo en el laboratorio de informática, donde cumplen tareas de MEP (Maestro de Enseñanza Práctica) de Informática, ATP (Auxiliar de Trabajos Prácticos) de Informática, AIE (Auxiliar de Informática Educativa) o equivalentes. En cuanto a la herramienta de recolección de datos, los ítems considerados en torno a ello son el 18, el 19, el 22 y el 25, e indirectamente, el 10, el 12 y el 20 (cf. Anexo 1: Entrevistas → Preguntas). Los indicadores consignados en dichos ítems son los siguientes:

- PROA, es decir, tratamientos de las urgencias (valor discreto, booleano), codificado como 1 (proactivo) y 0 (reactivo); y
- CTE (Complejidad de Tecnología Empleada) comprende dos categorías: a) presencia de herramientas *backend* (realimentadoras del CM, por ejemplo debido al orgullo de emplear las herramientas que estudió y que los hiciera sentirse aptos para mejorar la escuela, para salir de ella si les place); y b) ausencia de herramientas del lado *backend*, solo *frontend* (herramientas de usuario). En ambos casos se obtuvieron valores continuos con precisión determinada de acuerdo al siguiente cuadro:

	Palabras claves / Términos a buscar	Escala
PROA	No hay palabras específicas y se vuelven a revisar todas las entrevistas expresión por expresión.	[1,0]
CTE Herramientas <i>Frontend</i>	<i>Windows, Microsoft Office.</i>	+1
	Profundidad manifiesta en el uso de <i>Windows</i> y <i>Microsoft Office</i> .	+2
	Instalación, adopción o impulso de programas educativos: <i>Educaplay, Click, Logo</i> , etc.	
	Servicio técnico elemental y reemplazo de partes.	
	Empleo de ofimática alternativa ( <i>Google Drive, Libreoffice</i> ) o servicios en la nube.	
Análisis funcional		
CTE Herramientas <i>Backend</i>	Configuración de <i>routers</i> , administración de la red de la escuela, administración del <i>hosting</i> o de alguna VPS.	+3
	Unificación de datos dentro de motores de base.	
	Adaptación de programas liberados ( <i>Microsis, Alba</i> , etc.).	
	Adaptación de plataformas de colaboración ( <i>Sharepoint</i> y otros) o adaptación de CMS o de LMS ( <i>Joomla, Drupal, Wordpress, Moodle</i> , etc.).	
	Configuración de <i>Linux</i> o <i>Windows Server</i> .	
CLI ( <i>Powershell, Shell</i> , BAT, confección de <i>scripts</i> de <i>backup</i> , recabado de información, <i>scrapping</i> , volcados de datos entre distintas fuentes).		
Programación en algún lenguaje, uso de <i>frameworks</i> , confección de página web a medida de la institución.	+4	

De los valores prospectados se obtiene una tabla, cuyas casillas grises indican ausencia de valores, dando a entender que el entrevistado no cumple tareas de encargado de laboratorio, a saber:

N.º Caso	Indicadores	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	PROM
Orden DP en Atlas.ti (Revisar Anexo 2 en CD)		P1	P8 P9	P4	P2	P6	P17	P12	P13	P18	P11	P49	P14	P5	P15	P10	P3	
A - Analista en Sistemas	PRO	0	1	1								1					1	0,8
	CTE	1	1	1								3					2	1,6
L - Licenc. en Sistemas	PRO						0											0
	CTE						1											1
I - Ingeniero	PRO							0					1					0,5
	CTE							2					3					2,5
O <sub>T</sub> . Otros: Técnico en Redes	PRO								1									1
	CTE								3									3
O <sub>D</sub> . Otros Diseñador Industr	PRO																	-
	CTE																	-
O <sub>P</sub> . Otros: Programador	PRO				1													1
	CTE				3													3
O <sub>tu</sub> . Otros: Técnico Universitario	PRO					1												1
	CTE					3												3
P - Profesor en Informática	PRO													1				1
	CTE													2				2
P <sub>T</sub> . Profesor en Tecnología	PRO																	-
	CTE																	-
P <sub>F</sub> . Profesor en Filosofía	PRO																	-
	CTE																	-

El promedio final se hace por individuo, considerando el mayor título de grado alcanzado, o bien el título de grado similar. Al faltar suficientes muestras como para arrojar una discriminación por título, se realiza un promedio general, que arroja a continuación los siguientes rangos: PROA = 0,76; CTE = 2,3.

De los valores computados no se obtienen grandes sorpresas. Hay una proactividad manifiesta, por encima 0,5 sin llegar a 1, relacionados probablemente a compromisos altos, identificables por los valores de VD y de VE obtenidos en la hipótesis anterior. Este valor cuantitativo coincide con las entrevistas, en los cuales y a pesar de los sinsabores, la mayoría de los casos prefiere el trabajo en las escuelas.

En cuanto a los valores de CTE, estos se encuentran más atados al 2 que al 3. Estos valores cuantitativos coinciden con los datos cualitativos obtenidos en las rudimentarias observaciones etnográficas referidas al acometimiento de las tareas utilizando ofimática en forma muy competente, algo de servicio técnico, la presencia de un sistema muy completo a través de la implementación de *Access*, e, incluso, el detalle de una tarea de análisis funcional realizada en una misión especial en Dirección General de Escuelas como apoyo a un programador de *Genexus*. Además aparecen dos situaciones en las cuales se abren los archivos con *Openoffice* y los *pendrives* con *Linux*, pero por razones de lo más originales, para prevenir infecciones de virus.

Estos valores tienen una relación con el CM según el tamaño de escuela donde se desempeña el entrevistado. En caso de escuelas con pocos cientos de estudiantes, no está acompañado de valores bajos de CM, lo cual es esperable: un paquete ofimático, más algún programa provisto por el gobierno (administración de *netbooks*, analíticos) son suficientes para resolver la mayoría de las situaciones. Lo mismo ocurre con escuelas privadas, donde contratan algún "enlatado" con la mayoría de las funciones previstas.

El problema se suscita en establecimientos educativos estatales de gran envergadura poblacional, tales como la escuela 4-083 "Agustín Álvarez" o la Escuela Normal 9-002 "Tomás Godoy Cruz". En estos casos se pudieron encontrar "estados espirales", que pueden resumirse en el círculo entrópico de

datos ya comentado: *dificultad para cambiar* → *estado de urgencia* → *sin tiempo para cambiar*.

Al cabo de varias iteraciones de este bucle, la consecuencia se puede observar en las propias entrevistas, en las cuales se advierte resistencia paulatina al cambio, algo de estrés producido por la acumulación de tareas, la cabal comprensión de estar atrapados en las escuelas, la timidez gradual y, finalmente, la timidez absoluta, es decir, de no poder ingresar en las empresas tras una carrera jalonada "solo" en el ámbito educativo.

Posteriormente y desde una perspectiva terapéutica, se retomará el problema de los bucles y de los cambios de nivel, pero en este punto se desprende el único consejo posible a los trabajadores actuales en los laboratorios, como una prioridad, tanto para la escuela como para su propio desarrollo profesional: si no desean terminar en un bucle semejante, deben lograr que les sean concedidos horarios para automatizar. Si no cuentan con el apoyo de los directivos, deben convencerlos mediante maquetas o *mockups* mínimamente funcionales, tales que interesen a todos los involucrados.

Respecto de los directivos, se debe legalizar que se usa al informático para tareas administrativas escolares y que la solución de entrenar a los usuarios no funciona, por la frecuente razón de que el personal administrativo, los preceptores y los secretarios se niegan a concurrir. A modo de ejemplo, la sola explicación de la fórmula necesaria para hacer un cálculo de notas bajo un sistema logarítmico, que demanda en la práctica unas cuantas reglas de tres simples conectadas por unos condicionales ("SI, ENTONCES"), a menudo provoca el alejamiento de los usuarios. Algo semejante ocurre con la explicación de cómo traer información de otras fuentes, o bien combinar correspondencia, lo cual es esperable: estas acciones requieren al menos de buscar datos dispersos, desactualizados y repetidos desde múltiples fuentes.

Nuevamente, llevado a poblaciones escolares grandes, cuando la información crece, estas tareas tienden a ser arduas de configurar. Por ejemplo, la instrucción del equipo directivo de "enviar un correo a todos los padres" no es lo mismo en una escuela con siete cursos (29 alumnos x 29 padres (o tutores responsables) = 203 correos). Si el límite de envío de Yahoo es de 50 destinatarios, requiere de 5 envíos. En cambio, si hablamos de la Escuela Normal, que tiene un promedio de 1.300 alumnos y que junto al personal (docente, de preceptoría, administración y celaduría) asciende a 1.800 personas, se puede dar una idea del tamaño de la tarea si esta es manual, por no mencionar el mantenimiento de la base de datos: padres (cuyos hijos han egresado), gestión de correos rebotados, manejo de filtros *antispam*, etc., que requieren de un sistema del lado de la escuela y, posiblemente, de un servicio SMTP contratado externo. Esta es la típica situación en la cual, de estar concentrada la información en un solo lugar, toda tarea en la escuela, ya sea de envío de correos, de cálculo de notas o de control de personal, debería llevar no más de dos pasos para los usuarios del sistema.

El razonamiento indica que es preferible destinar al informático a resolver este problema de una vez por todas: asignarle las horas y controlar el progreso mediante alguna forma simplificada de "*desarrollo ágil*". Si el directivo va a utilizar al informático, lo hará explícitamente y de esta forma (cuidando de no infravalorar sus capacidades, impartiendo órdenes equivalentes a las que recibiría en una empresa o preparándolo para una instancia semejante). La forma más directa es considerar esta cuestión dentro de la formación del personal directivo de escuelas como una competencia para la gestión y la toma de decisiones.

Si uno de los problemas para acometer cambios de gran magnitud, como el caso del círculo entrópico de datos, es la falta de nociones de alguna teoría asociada al cambio (la Teoría General de Sistemas por diversas razones de

cercanía a la profesión, conexión de sus autores con la cibernética, orientación al constructivismo y, especialmente, por su acoplamiento y coherencia transversal a una psicología dedicada al tema del cambio), entonces resulta menester ocuparse de una formación en la gestión del cambio (reingeniería de sistemas organizacionales y de manejo de la información). A tal efecto se realizó a los entrevistados la pregunta 21 que decía: "¿Qué recordás de la Teoría General de Sistemas? ¿Qué aplicas de ella?". En los casos que se contó con ayuda de entrevistadores, se los instruyó para ahondar en la respuesta, indagando por autores y conceptos tales como 'entropía', 'realimentación de refuerzo', 'realimentación positiva', 'retroalimentación negativa', 'efecto de palanca', etc.

Del análisis de datos no se construyen gráficos ni tablas, por ausencia de base empírica. Como fuera predicho en el capítulo 2, subtítulo 2.4 "Mecánica en lugar de sistemas", profundizado en el apéndice C "El pensamiento estratégico necesario en las carreras tecnológicas" y propuesto en la hipótesis H<sub>2</sub>: "La Teoría General de Sistemas, así como otros modelos teóricos heurísticos asociados a la complejidad estarían ausentes en la formación de los estudiantes de sistemas", no hubo un solo caso en que los entrevistados recordaran la Teoría de Sistemas, más que el consabido esbozo de una caja negra, con una entrada, un proceso interno y una salida.

Respecto de la hipótesis H<sub>3.1</sub>, "La resistencia al cambio en los docentes de NTIC se debería a la presencia de "estancamiento profesional", rasgos neofóbicos, pensamiento algorítmico lineal que tiende a esclerosarse, escasa investigación, sostenimiento de verdades absolutas, descarte de otras fuentes de información solo por encontrarse éstas en otros idiomas y enmascaramiento de su limitado dominio sobre el cambio tecnológico bajo una representación social de aparente actualización", la misma no resulta sencillo de contrastar, por revestir numerosos componentes (indicadores) que es preciso indagar.

El examen de estos componentes demanda un dominio táctico al momento de formular las preguntas dentro de las entrevistas, de modo tal que no alerten posibles suspicacias o falseamientos de información por parte de los entrevistados. El trabajo de campo resultó ciertamente arduo. Ello requirió el establecimiento de un vínculo empático que permitiera a los entrevistados sentirse cómodos, tal que se abrieran a los entrevistadores, principalmente para plantear los ciertos indicadores, confiabilidad que se alcanzó plenamente en los casos P1, P2 y P7. Esta familiaridad reduce la calidad de la muestra en términos epistemológicos, y convierte el problema del muestreo en una caja de Schrödinger, donde es imposible presentar datos que no se encuentren alterados, y que no sean potencialmente correlaciones espurias producidas por el mismo instrumento de recolección, que es la entrevista. Así, pues, para resolver este problema de medición, se emplearon varias estrategias de análisis e interpretación de la información.

En primer lugar, la hipótesis se descompuso en varias preguntas cruzadas. En el "Anexo 1: Entrevistas" se puede revisar aquellas preguntas numeradas como 4, 7, 8, 10, 16, 17, 18, 19, 20, 22 y 23. De las mismas y su resultados se sabe, por ejemplo, que los informáticos se limitan a herramientas *frontend*.

En segundo lugar, para acompañar a las entrevistas y respecto especialmente de esta última observación, se incluye a aquellas realizadas a guisa de instrumento de recolección de datos, es decir, la observación desde un punto de vista etnográfico. Estas observaciones están desarrolladas en el Capítulo 3, y de ellas se obtiene, por ejemplo, que el problema no es solo que apenas usen herramientas *backend*, sino que ante un cambio en una herramienta *frontend*, tienden a dejar de usar completamente las computadoras.

No obstante, para cumplir con la rigurosidad científica necesaria, solo las observaciones cruzadas con otros investigadores pueden aportar la suficiente

confiabilidad sobre estos casos. Por esta razón se deja al final, como propuesta y a guisa de corolario, la petición al lector de realizar sus propias observaciones, de forma tal que estas puedan contrastar la hipótesis. La única condición al respecto es realizar las observaciones en condiciones similares: escuelas estatales con poblaciones escolares equivalentes: aproximadamente entre los 500 y los 1.500 alumnos. Solo los guarismos que se dan en estos ambientes ponen a prueba la capacidad de obtener datos y convertirlos en información, acaso la única noción teórica de sistemas que tienen clara y en común los entrevistados.

En tercer lugar y sin pretender la autoreferencia, se cuenta con las variables AC (Acceso a Complejidad) y BA (Baja Autoestima) medidas en la hipótesis H<sub>1.2</sub>

$$BA = 0,1428571429; AC = -0,1428571429$$

Al correlacionar ambos valores se aprovecha un modelo ya existente para construir una herramienta tal que relacione: a) el AC capaz de abarcar, datos que ya se tienen; y b) en cuanto AC, de qué tipo se les exige (disgregación necesaria). Luego, lo que busca esta hipótesis es examinar la RC, donde RC1 [Resistencia al cambio 1], permite inferir que entre a y b no están dispuestos a enfrentar motivaciones personales, pasatiempos, familia, distancia entre zonas e, incluso, razones políticas para no colaborar (por ejemplo: rechazo a *Conectar Igualdad*). En RC2 [Resistencia al cambio 2], la diferencia entre a y b que no pueden enfrentar, cuyo valor es equiparable a BA. Sin embargo, ¿cómo se distingue cual resistencia se está examinando? Bien, pues por la presencia de valores similares RC2 = BA. Esta aparece en RC2, porque a diferencia de RC1, los sujetos entrevistados no pueden "elegir" salir de la escuela.

La pregunta, asociada a esta última aseveración, bien podría ser, averiguar que hay respecto de una mezcla entre ambas. Cabe alegar que algunos individuos manifiestan ambas resistencias. Nuevas posibilidades de análisis se abren y se

ofrecen al lector. Por ejemplo, la relación entre RC1 y RC2, puesto que RC1, sostenido en el tiempo devendría en RC2, pues la resistencia *actitudinal* de no cambiar provocaría la resistencia *real* a cambiar por falta de ejercicio.

Respecto de la disgregación de AC, esta no puede dejarse de lado, puesto que no todas las personas pueden acometer cualquier tipo de complejidad. Personas muy competentes en un área, no siempre lo son en otras, sin hablar de disciplinas diferentes. De hecho, el factor del perfil estudiado en el Capítulo 1 podría estar incompleto en materia de "informáticos", y el rango comprendido entre "Analista", "Licenciado", "Técnico" e "Ingeniero" está completamente desfasado. Así lo entienden varios gigantes del *software* y compañías internacionales, quienes hacen caso omiso a estas titulaciones, puesto que prefieren tomar recursos humanos que evidencien certificaciones internacionales en herramientas que ellos usan. La razón de ello estriba en que el campo de la informática se ha especializado en varios tipos de complejidad. Pretender que un encargado de laboratorio pueda acometerlas a todas es potencialmente una ingenuidad. Por ahora baste con el siguiente ejemplo: si la escuela "x" necesita de un desarrollo de *software*, o quiere impartir nociones de mecatrónica, pero recibe en cambio un administrador de redes, éste, a menos que posea condiciones singulares, no podrá acometer la tarea. Por tal razón se busca un modelo ya aceptado, y usando esa base, en el siguiente capítulo se construye un instrumento nuevo, basado en escalas y exigencias.

Respecto de la H<sub>3.2</sub>: "El docente de tecnología, cuando se encuentra "atascado", presentaría resistencias al pedido de ayuda, vería afectada su autoestima profesional y tendría resistencia a seguir protocolos éticos cooperativos y colaborativos, si éstos lo ponen en evidencia", sucede lo mismo que con la H<sub>2</sub>. Se considera válida la expresión al menos en la población estudiada, la cual está referida no tanto al docente encargado de laboratorio, sino más bien al que



Un fragmento obtenido en una entrevista (cf. Anexo 1: Entrevista P5) manifiesta:

A: —Otros profesores de informática, ¿te preguntan?, ¿te consultan? Vos llevás muchos años en eso. ¿Son de venir y decirte: "Estoy bloqueado"?

B: —No, no, no, o sea, no profesores de informática, sino los otros.

A: —Pero los de informática podrían ser también, ¿o sos el único profesor en esas escuelas?

B: —Sí, en general soy el único. Acordate que se da en un solo año.

A: —¿Y profesores informáticos amigos o conocidos que tengas? A lo mejor no se ha dado la situación; es para medir el grado de colaboración que tienen.

B: —Yo colaboro con profesores. Me piden cosas o necesitan hacer algo, pero también con la limitación de que yo estoy en la escuela un tiempo y después me tengo que ir a otra. Cuando me consultan les doy una mano en ese aspecto, ahora de informática. Informática no, porque los tipos en informática la tienen muy clara, y la parte técnica la manejan bárbaro. ¿Sí? Entonces yo qué les voy a estar diciendo cómo armar una plataforma si ya la tienen clara.

A: —Yo he conocido varios profesores que no la tienen muy clara y me he acercado. "¿Cuál es tu problema?", "No entra a Internet", ¿Querés que te ayude?". Y no, no quiere.

B: —Sí, bueno... el que es bruto no te consulta. Es bruto porque no le interesa superarse. El que la tiene clara es un autodidacta y no te

consulta, porque él ya sabe cómo hacerlo. ¿Me explico? Me he encontrado con esa... con ese tipo de gente. Conozco a tipos brutos en un laboratorio, que le digo: "Necesito que ande la red"; y dice "No, no puedo. No tengo tiempo". Y... mentira. No sabe hacerla, pero yo tampoco tengo tiempo, entonces tiene *Internet* la sala de informática... están todas las computadoras conectadas a *Internet*, pero no puede conectar las máquinas en red. Entonces yo no me puedo poner a hacer eso porque no me dan los tiempos; además no es mi laburo. Mi laburo es ir a dar clases, tomar, evaluar y el tipo encargado es un nabo; y ese tipo no te pregunta en su puta vida nada. Los tipos que son buenos no te preguntan porque ya la tienen clara".

Se cita nuevamente la expresión "no profesores de informática, sino los otros", señalando también el hecho que interactúan con otros docentes en tanto estos no pertenezcan al mismo espacio curricular.

En el caso de los encargados de laboratorio, el problema se mide por ausencias de colaboración en foros *online*. Es decir, aprovechan las respuestas existentes, pero no participan ni colaboran, lo cual es una falta de ética a los protocolos de colaboración de los foros. Finalmente, preguntan personalmente a referentes de confianza por privado, que no poner en evidencia la ausencia de un conocimiento. Entre las citas posibles se escoge la siguiente:

23. ¿Qué haces cuando te encontrás con un problema de bloqueo de un programa?

B: —Ahí tengo justo uno [un caso]. Me voy derecho a *Google*. Y últimamente, me estoy yendo mucho a *Youtube*. Lo bueno es que los tutoriales ahí son en vídeo. Sí, ahí ya no lo resuelvo. Lo resuelvo preguntando.

A: —¿A quiénes preguntas?

B: —A conocidos que yo sé que saben, que me podrían llegar a apuntar.

A: —¿Tenés un círculo de personas referentes?

A: —Sí, tengo ahí unas "puntas".

24. ¿Participás actualmente de alguna lista de correos?

B: —No. Veo mucho en los foros, cuando voy a *Internet*. No... porque no tengo paciencia para esperar a la respuesta. En los foros, en las listas vos preguntás, y te contestan al otro día, o dos días después... si es que te contestan. Porque yo he visto que preguntan las personas, y que la respuesta está vacía. Y en las listas, la verdad que nunca publiqué nada (cfr. Anexo 1: Entrevista P2).

## Capítulo 6: Análisis de Datos: prospección de un modelo sistémico complejo de resistencia al cambio

A continuación se presenta el modelo escogido para analizar los datos.

Podemos bosquejar a las personas que se encuentran impartiendo tecnología e informática en las escuelas, casi independientemente de sus títulos habilitantes, en dos tipos, de acuerdo al *grado* y al *tipo* de complejidad que pueden manejar.

De esto ya había hecho una referencia respecto de utilizar niveles de *profundidad*, a la manera de un especialista en este tema, Carlos Reynoso, a quien vuelvo a citar "En mi concepto, simplicidad y complejidad resultan de adoptar escalas, articular variables o definir focos en el plano epistemológico" (2011, pp. 25). Y pone de ejemplo: "mayores grado de libertad, mayores números de elementos, relaciones de alta densidad", etc.

Respecto de clasificar *por tipos*, mantengo el modelo de explicación escogido por Reynoso en sus obras, que no es otro que la idea de complejidad organizada descrita por Warren Weaver (1948). Este autor distingue los tipos de acuerdo a la siguiente tabla:

<b>Modelo</b>	<b>Perspectiva del objeto</b>	<b>Inferencia</b>	<b>Propósito</b>
I. Mecánico	Simplicidad organizada	Analítica; deductiva; determinista; cuantificación universal	Explicación
II. Estadístico	Complejidad desorganizada	Sintética; inductiva; probabilista; cuantificación existencial;	Correlación
III. Complejo o sistémico	Complejidad organizada	Holista o emergentes; determinista; cuantificación conforme a modelo	Descripción estructural o procesual; modelado dinámico
IV. Interpretativo o discursivo	Simplicidad desorganizada	Estética, abductiva; indeterminista; cuantificación individual	Comprensión

A la manera de Reynoso, quien señala que no hace falta utilizar todos los tipos, es que se procede a clasificar solo bajo los tipos I y III.

Siguiendo este modelo de explicación, se presentan en primer lugar personas con habilidad para abarcar procesos mecánicos, con tendencia a explicar las situaciones usando este prisma: descomponen un problema en partes y presuponen que estas tienen comportamiento determinista. Estas partes pueden ser tornillos, transistores o alumnos, en el sentido de instancias de clases Usuario.

Independientemente de esta observación, y acompañando la idea general, en programación y en redes, a medida que un lenguaje o que una comunicación adquieren especificidad, y se encuentran más cerca de la materia física o del *hardware*, se los considera más de "bajo nivel". Si para un psicólogo se ha apelado a la analogía de hemisferio izquierdo y derecho, se utilizará para lingüistas y para informáticos versados en desarrollo de *software*, la distinción "arriba" de "abajo" o "altos" y "bajos", sin que esto indique un "mejor" o "peor" en lo absoluto.

Primeramente describimos a aquellas personas con facilidad para desarmar problemas siguiendo el modelo mecánico o de tipo I. El nivel de complejidad al que se enfrentan será ordenado de acuerdo a su mayor número de elementos, esto admitiendo que dejamos de lado sus relaciones de alta densidad, ya que no siempre es posible calcular su número<sup>51</sup>.

1. *Destreza combinando capas de software que están a nivel alto, de aplicación o "de usuario"*: configurar programas, instalarlas, quitar virus, entre otras tareas. Normalmente un usuario no maneja más 10 programas a la vez, y por lo tanto este será nuestro número de referencia respecto de los siguientes niveles;
2. *Destreza combinando piezas mecánicas*: típicamente, reemplazo de partes o servicio técnico simple. Una computadora hogareña por ejemplo puede tener aproximadamente 20 elementos: teclado, monitor, placa de video, placa madre, etc. El problema se solucionará cambiando algunas o todas las piezas;

51 De incluir a este ordenamiento de complejidad las relaciones entre los elementos considerados, necesitaríamos computar diversas combinaciones que nos llevarían aquí varios tiempos además del constante: del doble exponencial, pasando por el logarítmico, el lineal, el cuadrático, el cúbico, al factorial, entre otros (Reynoso, 2011, pp. 15).

3. *Destreza configurando piezas de software que están más cerca del hardware:* sistemas operativos, *runtimes*, máquinas virtuales, etc.; en este caso, al entendimiento de las piezas anteriores debe sumarse por ejemplo en *Windows 7*, 46 procesos de Sistema Operativo, del tipo que se coordinan para atender al usuario y al *hardware*; el informático debe entender el funcionamiento de varios de estos procesos para distinguir que la falta de comunicación con la impresora se debe, por ejemplo, al Servicio de Impresión, y no al cable USB;
4. *Destreza comunicando equipos (redes y comunicaciones):* hay muchas enumeraciones posibles, sumando equipos y protocolos; si se sigue con la idea de medir complejidad, agregar más elementos y sin hacer una enumeración completa en este nivel, entonces basta mencionar que solo el protocolo TCP/IP provee nominalmente 65.535 conexiones a servicios —y TCP/IP es solo uno de los tantos protocolos existentes;
5. *Destreza combinando piezas desconectadas de software que sirven para crear software nuevo* (programación de código, compiladores e injertos de librerías creadas por otras personas): la cantidad de elementos aquí es muy difícil de prospectar, no solo porque existen cientos de lenguajes de programación y porque cada uno de estos lenguajes cuenta con cientos de librerías "base", sino porque cualquier programador puede definir librerías nuevas y, en el caso de los lenguajes libres, crear nuevas versiones de los mismos; y
6. *Destreza creando o combinando hardware, llegando al límite de la materia, según leyes lógicas, matemáticas y propiedades físicas:* en este nivel de complejidad, el número que resulta apropiado traer a colación no es el de los componentes electrónicos (cuya cantidad de combinaciones pueden ubicarse en niveles anteriores), sino los transistores de silicio; el equipo de

ingenieros que ha trabajado, en el núcleo de un procesador construido en 2015, ha debido combinar las compuertas (flip / flop) de 20.000 millones de transistores, cuidando que no se fundan térmicamente entre sí<sup>52</sup> —y aquí no se incluyen siquiera las promesas de la mecánica cuántica.

Por otro lado, en el tipo III, se tiene a personas capaces de abstraerse de tales mecanicidades (o inversamente, incapaces de sumergirse en ellas). El modelo que siguen para estudiar sus objetos es de tipo sistémico. Este se visibiliza a través de la habilidad para elevarse por sobre los detalles, a fin de capitalizar la dimensión holística del conjunto y detectar sus emergentes. De acuerdo al análisis de datos, la mayoría de los sujetos entrevistados infieren lo desconocido siguiendo este modelo. De acuerdo al nivel de complejidad que pueden abstraer, su habilidad se puede ordenar siguiendo la escala que figura a continuación:

1. *Destrezas simples, típicamente empleadas en tareas administrativas:* debido a que la mayoría de los docentes informáticos de las escuelas manejan solo este nivel de destrezas; los contenidos que se brindan a los estudiantes son de esta clase; los alumnos generan una mentalidad "administrativa", que activa bucles de realimentación; raramente un usuario mezcla más de 10 programas a la vez, pues prefiere a aquellos con enlaces fuertes y obvios, como las *suites* ofimáticas;
2. *Destrezas ofimáticas avanzadas en combinación con su entorno:* refiere a habilidades de desarrollo de *macros* o combinación de piezas de *software* con enlaces aparentemente débiles; sus practicantes distinguen que Internet no es solo la Web (conocen de protocolos, saben convertir entre tipos de archivos, identifican y aprovechan estándares, entienden la utilidad

52 Anónimo (s.f.). IBM logra un avance clave en la fabricación de microprocesadores.

Recuperado de <http://www.lanacion.com.ar/1809280-ibm-logra-un-avance-clave-en-la-fabricacion-de-microprocesadores>, [consulta el 14 de julio de 2015].

de los archivos CSV o de los códigos QR, saben armar foros, aulas virtuales y, especialmente, aprovechar las ventajas de la globalización); una grosera cuantificación de elementos a usar como ejemplo es una campaña de *e-mail marketing*, en la cual el usuario deberá evaluar varias plataformas, considerar los protocolos necesarios [SMTP o API], habituarse a un cliente de correo, manejarse con plantillas web, obtener direcciones de correo, limpiarlas, atenerse a reglas de etiquetas de Internet, respetar los filtros de los robots, etc., para todo ello deberá, en resumen, combinar al menos 20 artefactos tecnológicos para lograr su cometido;

3. *Destrezas avanzadas en herramientas online*: en este nivel de complejidad se encuentran usuarios capaces de aprovechar las nubes, implementar técnicas SEO, realizar campañas de *marketing* en redes sociales, "mashear" servicios, identificar negocios que se pueden construir sobre otros negocios, entender la utilidad de las API, de los *web services* o de metodologías *scrapping*; estas personas son emprendedores, generadores de empresas *startups*, que saben detectar nichos de mercado por todas partes; si además tienen alguna necesidad de las planteadas anteriormente, saben como definirla claramente, delegarla o encargarla para su ejecución a *freelancers* o a personas con nivel 5 en el tipo I —desarrollado anteriormente—, liderando si hiciera falta; esta destreza requiere de buenas habilidades sociales, manejo de inteligencia emocional, e incluso buen gusto y olfato para descubrir tendencias. Un ejemplo contundente es el del ingeniero social, el tipo de *hacker* más frecuente que hay en Internet, el cual en lugar de crear un elaborado mecanismo determinista para robar una contraseña, simplemente se la pide al usuario. La complejidad aquí es difícil de medir, ya que no es posible despreciar la cantidad de interacciones entre

los elementos, pues la mayoría de ellos ya no son artefactos sino personas; se trata de un terreno tanto matemático como sociológico<sup>53</sup>.

Tras las consideraciones precedentes el problema abordado en esta tesina, acerca del papel que tienen los informáticos en las escuelas, el mismo puede ser medido según la exigencia que se les demanda de cada clase. Específicamente, el conflicto se produce en el nivel de las solicitudes implícitas, es decir, cuando se le exige a los profesionales una capacidad "todo terreno" o "a todo cambio". De este modo cabe plantear la siguiente distinción, a saber:

- 1) *exigencia de dualidad*: la capacidad de dominar tanto destrezas de tipo mecánico como de tipo holístico;
- 2) *exigencia de nivel 4 a nivel mecánico*: dada la cantidad de trabajo técnico que hay en una escuela, se pretende que el docente de informática pueda al menos ser competente en la manipulación de objetos de cuarto nivel; y
- 3) *exigencia de nivel 2 a nivel sistémico*: cuando se le exige que en sus soluciones sistémicas incluya al menos un segundo nivel de análisis.

Dominar este conjunto de niveles requiere de capacidades psíquicas notables. Aquello que San Agustín refiere en cuanto a los dones propiamente humanos, esto es, la inteligencia y la voluntad, ambos en equilibrada proporción. Sin embargo, en estos niveles, aquello que se denomina como "resistencia al cambio" y "resistencia a la complejidad" es, *stricto sensu*, un límite físico impuesto por el cerebro, un órgano que se bloquea al seguir un mecanismo tanto de

53 Hasta donde se pudo constatar, las únicas obras que tratan sobre el tema son *Complejidad y Caos* (2006) y *Redes Sociales y Complejidad* (2011) de Carlos Reynoso.

conservación de energía como de control homeostático regionalizado. Este es un terreno de la genética, la física y la neurociencia, quienes explican de diversas formas como el cerebro ante "el dolor" de lo inabarcable dispara mecanismos de bloqueo, tales como mecanismo de defensa o protección (autoconservación). La comprobación de estas palabras es innecesaria, puesto que solo basta con proponer al lector la realización de un ejercicio de abarcamiento de complejidad, que consiste en lo siguiente: si se le solicita a alguien la resolución completa, concreta y final de cualquier pregunta fundamental, tales como "¿existe Dios?", "¿tiene un fin el universo?" o "¿hay un universo afuera del nuestro?", ello fácil y rápidamente lleva a un colapso explicativo por falta de abarcamiento del objeto y por ausencia de vías (métodos) para su tratamiento especulativo. Incluso si todo el esfuerzo residiera en construir una respuesta y esta estuviera puesta de manifiesto, el procesamiento necesario superaría la capacidad que tiene el cerebro para alimentarse en el tiempo hasta obtener un resultado concluyente y disipar la cantidad de calor que tomaría dicho cálculo. Es decir, ello superaría su límite homeostático, incluso con el auxilio de máquinas. La respuesta a esas preguntas quedan limitadas al terreno de la especulación científica, con un amplio margen de incertidumbre, al mejor estilo de Isaac Asimov y su serie Fundación, el único autor que se ha atrevido a jugar con la idea de predecir el destino de la humanidad mediante la matemática y la antropología.

Al considerar este límite material, ahora cabe describir la incomodidad de los practicantes de tipo I, esto es, de los mecanicistas. Ello se pone de manifiesto cuando se ven obligados a simplificar en el aula y crear imágenes completas *de/total* de las piezas en las cuales ellos son especialistas, así como cuando deben dejar de lado su papel de especialistas para ser generalistas o sistémicos.

A guisa de semblanza es fácil evocar la incomodidad de las personas acostumbradas a las abstracciones mecánicas, al atenerse a comportamientos

sociales y detalles estéticos para presentar sus algoritmos. Análogo a físicos, lingüistas y matemáticos que deben cuidarse de resumir sus expresiones sin cercenar los detalles, cuando deben presentar sus ideas a otras personas en forma de paquetes terminados y disfrazarlas como "simples" para poder "venderlas". Esta ingrata y exigente dualidad *full stack* bien la conocen los programadores de aplicaciones web, a quienes se les exige poseer ambos conocimientos de *backend / frontend*, para construir "*interfases* enriquecidas". Tal es la exigencia moderna de contar con toda clase de elementos atractivos que mantengan la atención del cliente y que no desalienten el consumo del producto.

En cuanto a los sistémicos del tipo III, quienes trascienden el problema desde un punto de vista holístico, de percepción de conjunto, cuando se ven obligados a resolver cuestiones mecánicas, niveles de la escala desarrollada a las cuales normalmente no descienden. Los jefes de sistemas y proyectistas se hallan enmarcados en esta tipificación, pues su capacidad yace en encontrar recursos humanos, hábiles y mecánicos, y lo cual determina en definitiva su grado de supervivencia y asertividad. Ello es normal en el terreno empresarial, tanto como lo es inusual en el aula (los docentes y los encargados de laboratorio no pueden "escoger su equipo"; en todo caso forman parte de un equipo donde el directivo, como se había señalado, raramente es capaz de enumerar, y menos aún de especificar los requisitos del producto para un artefacto tecnológico).

Este problema de ser dualmente competentes, es un reflejo de los *tiempos exponenciales* que vivimos, usando la expresión de Fisch, K. Y McLeod, S. (2012), en el video educativo "*Shift Happens*". La pasión por el detalle ya no es un rasgo deseable: es una obligación del puesto de trabajo, y esto puede comprobarse observando los artefactos en los que debe ser competente el usuario. Antaño (en tecnología, hace apenas unos 10 años) se requería de dispositivos perfectos. Hoy en cambio, raramente los programadores ofrecen predicciones certeras, respecto

de la estabilidad de sus *softwares*. Se *acepta* honestamente la incompletitud permanente, y se blanquea el tema cuando se negocia la programación de un *software* mediante técnicas conocidas como *desarrollo ágil*. El usuario acepta las actualizaciones constantes del sistema operativo y de los programas: así lo ha empezado a acostumbrar su propio celular, para pasar a aceptar sistemas morfogénéticos en lugar de los anteriores morfoestáticos. El usuario acepta, y lo mismo debería hacer el docente de informática, que merced aumentan la complejidad de los mecanismos, estos pueden fallar no solo a nivel mecánico, sino que también pueden llegar a producir toda clase de emergentes no deseados.

Baste para ello con observar los fracasos de productos como los robots japoneses parecidos a humanos, que caen en la zona de incomodidad conocida como "el valle inquietante", o los lentes *Google Glass*, los cuales fueron rechazados por el mercado de consumo, por no predecir, por ejemplo, que a las personas les incomoda que las puedan estar filmando con los lentes o que se las pueda *googlear* en tiempo real.

Cuando el docente es conminado tanto a descender a niveles mecánicos profundos, como a ascender a niveles altos y contemplar sistemas desde afuera a fin de predecir sus emergentes, se lo pone en situaciones análogas a las de los científicos urgidos por crear una teoría unificada del todo.

En pocas palabras, para cambiar de rango tipológico es menester reconocer los "escalones" de la complejidad considerada. En esos escalones es donde las personas tienden a quedar atrapadas, en procesos reverberantes. Existen buenos programadores pero malos docentes, o, inversamente, docentes que enseñan bien ofimática y explican exitosamente los alcances de la web, pero que son incapaces de arreglar una conexión a Internet.

Si se retoma la hipótesis  $H_{3.1}$  que había propuesto que la resistencia al cambio puede obtenerse definiendo correctamente el AC (Acceso a la Complejidad) —ahora descompuesto en dos partes (mecánica y holística), donde a su vez se habían definido asimismo a y b (siendo "a" el AC capaz de abarcar datos que ya se tienen y "b" en cuanto AC, el tipo que se les exige —disgregación ya efectuada en la clasificación de tipos I y III—)—, entonces ahora se tiene formalmente una clave de resolución aceptable del problema, a saber:

- RC1 [Resistencia al cambio 1]: el restante entre "a" y "b" que no están dispuestos a enfrentar, debido a motivaciones personales, pasatiempos, familia, distancia entre zonas y razones políticas para no colaborar; y
- RC2 [Resistencia al cambio 2]: el restante entre "a" y "b" que no pueden abarcar cognitivamente.

También se había propuesto que la forma de distinguir RC1 de RC2 es por la presencia de valores similares ( $RC2 \cong BA$ ), siendo BA "Baja Autoestima". Esta aparece en RC2, porque a diferencia de RC1, el sujeto no puede "elegir" salir de la escuela.

De esta manera se proponen dos enfoques para resolver el problema.

En el caso de RC1, o "los que pueden, pero no quieren", que es la manifestación concreta de la negación a colaborar, la metodología a seguir empieza por parte del directivo, quien debe capacitarse en primer lugar como se sugiere en  $H_{1.3}$ , a fin de impartir las órdenes de forma correcta, en el lenguaje que un informático espera recibir. Como dice Antoine de Saint-Exupéry en *El Principito*: "si quieres que te obedezcan, imparte órdenes justas", o al menos que el informático entienda como justas. Seguidamente, realizar el blanqueo de la necesidad institucional del informático: la escuela "necesita" del informático. Lo

necesita por encima de sus deberes normales, al punto que es válido ponerlo en jaque bajo cualquier equivalente a juramento hipocrático que venga al caso, para solicitar acciones, que por supuesto no forman parte de las funciones del docente de tecnología o de informática, y que satisfagan a todas las partes.

Así, poner en vigencia que las tareas asignadas no degradan al informático si la complejidad es alta, sino que contribuyen a la carrera personal del mismo. En todo caso, degradarlo es ponerlo a hacer tareas rutinarias, lisiando su capacidad de optar algún día por desempeñarse en las empresas o en el *freelancing*. Legalizar, además, que la prioridad máxima está en los alumnos, para no caer en situaciones de máxima resistencia (como en el caso del equipo del área de Informática de la Escuela Normal 9-002, en la cual, de acuerdo a la cultura institucional establecida desde hace 10 años, la prioridad reside en "hacerle el trabajo" a los administrativos, por encima de los preceptos de *Conectar Igualdad*). Finalmente, impulsar en los especialistas que dictan informática y tecnología algún plan de carrera pedagógico (analistas licenciados, técnicos, ingenieros), tanto como a los profesores de informática impulsarlos en especializaciones que rellenen lagunas en el terreno técnico. Y en ambos casos, enaltecer sus trabajos enviándolos a congresos de encargados de laboratorio, fomentando proyectos escolares y participación en olimpiadas de informática tales que dignifiquen sus *curriculums*. De acuerdo a lo descubierto en las entrevistas, proyectos que fueron frecuentes en los años 90, hasta los problemas ocurridos en el país en el año 2001.

Respecto del tratamiento de RC2 ("los que no pueden"), el tema se resuelve mediante una combinación terapéutica. Inicialmente se planteó la enseñanza de técnicas de pensamiento científico como línea de acción, pero luego se descartó por ser demasiado ambiciosa en su aplicación inmediata, pues es raro que los informáticos sean científicos o filósofos. Acometer el aprendizaje de las ciencias

como objeto de estudio en sí mismo, resulta innecesario, por cuanto los informáticos ya poseen su propia herramienta para tratar problemas, rodeada de suficiente paradigma epistemológico y metodológico, a saber: la Teoría General de Sistemas. Su propuesta de inclusión en los programas de estudio ha sido ya desarrollado en el Capítulo 2.

En todo caso, tiene más sentido revisar aquellas herramientas que tienen en común la ciencia, la informática y la tecnología. Por ejemplo, en numerosos trabajos tecnológicos e informáticos es preciso recurrir a técnicas de procesamiento de datos (fase heurística). Sin ir más lejos, en temas de fichaje documental (que no tiene equivalente consensuado entre los informáticos), mediante el uso de *Zotero*, es posible guardar información relevante, procesarla, accesibilizarla, indexarla y recuperarla con gran rapidez. Lo mismo ocurre con el lenguaje R, un lenguaje propio de universidades e institutos de investigación, pero accesible a cualquier profesor de nivel Medio y de cualquier disciplina, para incentivar a los alumnos en el despliegue visual de funciones y gráficos estadísticos. El docente de informática solo debe seguir unos pocos tutoriales gratuitos para poder apoyar el estudio de las otras materias. Así rodearlas de un marco lúdico formidable, con la ventaja de poder incluir en su cátedra rudimentos de algoritmos aplicados a situaciones concretas.

Yendo específicamente al tratamiento de RC2 (relativo a la complejidad en su ejercicio), ésta resistencia es tratable si se centra el problema en la gestión del cambio, antes que estigmatizar la resistencia en sí misma. Y si se aborda el cambio como objeto de estudio en sí mismo, nadie mejor que uno de sus precursores —Paul Watzlawick, continuador de la TGS y la cibernética de segunda generación— quien aplica terapéuticamente la gestión del cambio sistémico-cognitivo (cibernético) en quienes no pueden *cambiar de nivel* por sí mismos. En este caso, las técnicas cibernéticas son aplicables tanto a los profesionales que se

frustran, como a aquellos docentes que, con sus bloqueos, malogran las esperanzas que el sistema educativo y la sociedad pone sobre ellos. En tanto que en el Capítulo 2, apartado 2.3 "Deficiencias en el abordaje de la complejidad durante las carreras de sistemas" se trata este tema, se sugiere revisar la obra completa de este destacado psicólogo sistémico<sup>54</sup>.

54 Cfr. *ut infra* Bibliografía.

## Conclusión

Un largo camino se ha recorrido desde la pregunta original, la cual indagaba respecto de la resistencia generalizada al cambio en docentes de informática y tecnología en relación a alumnos y profesores de otras áreas, quienes lo aceptaban sin mayores polémicas. Durante el desarrollo del trabajo se presentaron identificadores lo más objetivos posibles.

Al comenzar el trabajo, el objetivo consistió, y a riesgo de caer en una posición subjetiva, en determinar si había una resistencia específica de los docentes de informática y de los encargados de laboratorio hacia el software libre. Este punto pudo desarrollarse, pero no de la manera esperada originalmente. Al hacerse cargo de la asesoría del trabajo, el director de tesis, fue categórico en señalar que las resistencias forman parte del comportamiento humano y, que existe excelente teoría de base que trata el tema, dentro de lo que se conoce como Cambio y Complejidad. Se constató asimismo que no hay aplicación de estas teorías a los casos observados, de modo que el problema de investigación terminó por delimitarse al formar parte de un fenómeno que ha sido investigado de forma general en el terreno de la sociología, pero no aplicado de una forma específica (antropológica), es decir, aplicado a grupos humanos específicos como el que nos compete en este trabajo. De esta manera el camino cambió sustancialmente al tratar de entender en forma mas completa a los sujetos de estudio, antes que asumir una posición parcializada.

Buscando el origen de las resistencias, se planteó indagar que relación había entre la forma en que se enseña tecnología, tanto en la escuela secundaria como en los ciclos posteriores. Así se constató que los planes de estudio de escuelas secundarias son los mismos de los años 90, y que a diferencia de lo impartido en aquella década, no se trabaja la interfase por línea de comandos de manera tal

que trasunte la dificultad que cabe esperar encontrarse en casos de proseguir estudios superior o universitarios. Se encontró ausencias también de enseñanza de algoritmos, y de un enfoque exclusivamente ofimático, con algo de Web (distinguir de Web de Internet, que no se enseña). La falta de aportes tecnológicos a la escuela, en forma de automatizaciones o de construcción de algoritmos también es evidente entre los encargados de laboratorio, quienes al menos en las escuelas grandes, no aportan o no consiguen los espacios para construir mecanismos que procesen datos, dependiendo de pobres implementaciones otorgadas por el gobierno.

Respecto de ésta última afirmación, y considerando que cada año cursado en sus carreras por los docentes de informática y los encargados de laboratorio han tenido programación, base de datos o análisis de sistemas, es llamativa la poca aplicación que hacen de estas herramientas en los espacios escolares.

Por estos descubrimientos, se desgranó el objetivo original, para pasar a detectar si estas falencias tienen una relación con la forma en que les fue impartido tecnología originalmente.

Siguiendo un hilo sistémico y constructivista, en base a Barthes (1957; 2009, pp. 52), Ashby (1965), Krashen (1985) y Bruner (2004, pp. 104), Gómez (2002, pp. 153), todos citados oportunamente, y realizando las entrevistas a los sujetos, se pudo constatar que la resistencia al cambio se encuentra fuertemente condicionada al exceso de interfases gráficas facilitadoras, inhabilidad con línea de comandos, ausencia de software libre, ausencia de protocolos de cooperación relacionados, de unificación de datos, post procesamiento o permeabilidad con el exterior de la escuela. Todas las soluciones se encuentran centradas en el paquete ofimático de Microsoft, y los profesionales cumplen el rol de usuarios; *usuarios avanzados*. Como tal los entiende la política educativa actual, y así los

tiende a reubicar, como ayudantes de otros docentes, para descontento de los sujetos de estudio.

También hay que mencionar que si bien los docentes y los encargados forman parte de su desplazamiento, también se detecta ausencias en la formación de los directivos respecto de la gestión de recursos humanos técnicos; los directivos no se encuentran en condiciones de aprovechar la formidable oportunidad que supone tener informáticos a cargo, los cuales son normalmente relegados a tareas menores.

Tras lo cual, surge el objetivo de determinar si se adjuntan factores de resistencia al cambio durante los mecanismos de enseñanza de tecnología e informática, y se advierte que en efecto, la transmisión es directa, inductiva, y finalmente volitiva. No hay tratamiento específico para resolución de temas complejos. Solo matemáticas, lo cual es beneficioso por cuanto ésta permite abarcar objetos abstractos de estudio, pero insuficiente por cuanto la teoría que tienen los informáticos para tratar complejidad, es decir su propia Teoría de Sistemas, no se imparte incluso en las propias carreras de sistemas.

De acuerdo a lo prospectado, basta con oponer suficiente voluntad para aprobar los planes de estudio de ciclos superiores o universitarios. Una vez en las escuelas, varios de los entrevistados se manifestaron verbalmente desinteresados o inhábiles en los temas planteados, contenidos todos aprobados y en teoría asimilados en su carrera. Los perfiles publicados en las carreras y los pre universitarios también tienen relación, puesto que se constató incluso casos de arrepentimiento por haber emprendido estudios de sistemas.

De esta manera se puede afirmar que en efecto, se adjuntan factores de resistencia al cambio durante los mecanismos de enseñanza de tecnología e informática en el ciclo superior y universitario, que el problema comienza en el

secundario, y que todo forma parte de un ciclo cuando estos profesionales vuelven a ejercer al nivel medio.

Estas respuestas sirvieron para enmarcar otro fenómeno observado, consistente en describir los condicionamientos que tienen los docentes de Tecnología y de NTIC, respecto de aquellos docentes de otras áreas que emplean NTIC. En efecto, se descubrió que los conocimientos técnicos son especialmente condicionante para sus practicantes, y que la capacidad de abrazar el cambio que éstos deben oponer ha de ser mayor a la media. Tal afirmación se basa en la buena capacidad que tienen docentes de otras áreas para cambiar de interfase cuanto menos se desempeñen con otras, y explica el comportamiento según el cual ante una migración de computadoras a Linux, los docentes de informática y los encargados de laboratorio acusaron mucho más el cambio que sus colegas docentes de otras áreas.

Tras diversas entrevistas se constató que el objetivo de determinar la resistencia al cambio de las *interfases* no solo aparece en primer lugar como rasgo distintivo, sino que esconden toda una gama de estancamientos, sesgos cognitivos, prejuicios, autismos y esclerosamientos del pensamiento.

Durante el trabajo se exploraron varios intentos de explicación. Aquellas respuestas más prometedoras conforman el núcleo del trabajo por saturación teórica y dirigen los esfuerzos de explicación, en tanto que otras líneas de investigación surgidas tanto del tratamiento de la teoría como del cúmulo de unidades de datos codificadas, se han organizado como apéndices. Estos apéndices, a guisa de corolario, proponen otras líneas de trabajo y profundizan varias observaciones iniciales relacionadas con el problema de investigación, sobre el cual además de describirse, se amplía y se propone diversos tratamientos para su resolución, de modo tal que se pueda ayudar tanto a los sujetos de estudio. De lo observado, estos sujetos son informáticos supuestos herederos de

teorías sistémicas, que no han podido conectar con sus padres fundadores, cibernéticos, antropólogos y psicólogos del instituto de Palo Alto. También son sujetos indirectos de estudio los directivos de las escuelas, quienes de no iniciar en los cargos con estudios de gestión, fallan especialmente en distinguir la forma en que se gestionan los recursos humanos del área informática.

Respecto de las hipótesis planteadas, se encontró evidencia respecto de marcadas equivocaciones en la elección de la carrera profesional por parte de los actuales docentes de tecnología e informática. Si no fuera por la necesidad de contar con un título habilitante para poder concursar en los primeros tres llamados en los concursos a cargos en instituciones educativas secundarias y superiores no universitarias, varios de los docentes en ejercicio podrían haber seguido un curso de operador en PC, antes que cursar una carrera completa. Entre los entrevistados, por ejemplo, surgieron incluso arrepentimientos de haber iniciado estudios de sistemas. Inicialmente se adjuntó este fenómeno a un problema en los perfiles publicitados en las carreras. Sin embargo pronto surgió el hallazgo de que los estudiantes son conscientes de la equivocación a poco de comenzar sus estudios, y que sorprendentemente a mitad de cursado ya tienen planeado arribar a las escuelas no como opción "b", sino como opción "a". Es decir, sin comenzar un profesorado en Informática como debería ser la secuencia razonable de un informático interesado en enseñar, probablemente influenciado por compañeros de trabajo que ya se encuentran trabajando en escuelas.

Posteriormente se detectó que los empleos mencionados en las escuelas operan inicialmente tanto como refugio económico, así como cobijo de ciertas complejidades de la carrera, esto es programar, analizar datos, matemáticas y otras materias troncales. Y se emplea la palabra "cobijo" señalando el simple detalle que ninguna de las materias señaladas fueron aplicadas por los sujetos entrevistados dentro de ninguna institución educativa. De lo cual se deduce que la

escuela, lejos de convertirse en un espacio de aprendizaje y de investigación —como lo fue inicialmente para el autor del presente trabajo—, tiende a cercenar, cuando no de lisiar, toda posibilidad posterior de insertarse en el mercado laboral profesional y en el mercado informal *freelance* (programación y diseño). Las únicas excepciones detectadas se encontraron en el campo del servicio técnico de PC; área por demás saturada de personas que no necesitan de títulos profesionales para ejercer, plagada de competencias desleales y de nula aplicación de los conocimientos aprendidos en la carrera.

Se debe mencionar que este fenómeno de anclaje (y lastre) en las escuelas, se encontró mayormente conectado a analistas y licenciados en Sistemas, en tanto que es menos frecuente en estudiantes de Tecnología, de tecnicatura en Redes y de Ingeniería; especialmente los dos últimos, quienes tienen mejor penetración en el mercado laboral empresarial. Examinar este fenómeno de cerca sirve para encontrar la razón por la cual Analistas y Licenciados tienden a presentar mayores resistencias, no solo al cambio, sino al tipo de trabajo docente en sí mismo. Sus expresiones de hastío y diversas protestas son más frecuentes que en el caso de las otras profesiones que se encuentran ejerciendo en el aula; en tanto que Ingenieros y Técnicos en Redes prácticamente pueden escoger libremente su lugar de empleo, los primeros mencionados (y especialmente los resistentes tipo RC2) se encuentran más limitados. De esta manera, la motivación definida como CM (Componente Motivacional) influye cuando desciende en los resultados obtenidos. Máxime en los casos estudiados que dependen de la escuela para poder subsistir como única fuente de ingresos. Se debe señalar que la población estudiada no representa el total de casos posibles, es decir, no se analizaron *todos* los docentes de Mendoza, y en el caso que así hubiera sido, estos deberían compararse también con otras poblaciones.

Si se continua con lo relativo a la ausencia detectada en cuanto a la falta de aplicación en la escuela respecto de lo aprendido en estudios cursados en la universidad o ciclo superior no universitario, no se pudo detectar tampoco casos concretos de construcción de algoritmos facilitadores para la escuela, ni concentración de información mediante motores de bases de datos —solo uso de Access— ni reciclado de proyectos libres de gestión de escuelas, ni enseñanza de programación a los educandos, dentro o fuera de programa. Se descubrieron unos pocos posteos y consultas sobre el tema en los foros de educar, pocos en relación a la cantidad de docentes de informática a nivel país —al menos varios miles— de modo que la inquietud por enseñar temas fuera de programa podrían considerarse casos excepcionales y aislados.

El impacto sobre los alumnos es equivalente a estas resistencias, y se trabajó revisando la manera con que docentes desalentados imparten contenidos a los educandos, los cuales ya se hallan obsoletos o notablemente desactualizados, de escasa complejidad y rudimentarios (excesivamente simples) en materia de informática, a tal punto que los mismos educandos inician más tarde estudios de sistemas bajo la creencia de que la vida del informático es especialmente sencilla.

Los profesores parecen ganarse la vida munidos de someros conocimientos acerca de *Windows* y de *Microsoft Office*, mientras que omiten la enseñanza de otros procesos de *backend*. Los programas de estudio permanecen inalterados, y llevan sin cambios tantos años como los planes de estudio de la Facultad de Derecho de la Universidad de Cuyo<sup>55</sup>. Los docentes de informática enseñan con el mismo plan de estudios que se usaba en la época de MSDOS, al cual solo le han

55 Anónimo (s.f.). *Derecho renueva su plan de estudios por primera vez en 25 años en la UNCuyo*. Recuperado de <http://www.diariouno.com.ar/mendoza/derecho-renueva-su-plan-estudios-primera-vez-25-anos-la-uncuyo-20160521-n793212>, [consulta el 22 de mayo de 2016].

agregado la Web y ni siquiera han introducido temas tan relevantes como las redes LAN o *Internet*, porque ignoran en algunos casos la diferencia entre los tres conceptos. La paradoja que surge es que, si enseñaran el antiguo MSDOS, los alumnos tendrían una perspectiva más clara de la informática y de sus posibilidades.

Así, pues, un aspecto derivado del fenómeno retardatario de la actualización de los contenidos posiblemente traerá como consecuencia que el alumno de nivel secundario ingrese más tarde a la carrera de sistemas, confiado en que "las computadoras deben ser simples", cuando en realidad no lo son. Esta inexacta creencia va acompañada de aquella que sostiene que los problemas de una computadora se resuelven "reiniciando o formateando", ello sumado a otros tantos "atajos" aprendidos en clase. Este tipo de prácticas aplazan los problemas en lugar de resolverlos. Los problemas se tratan como "caprichos de la máquina", como si esta tuviera mecanismos volitivos. Y, en lugar de convertirse en oportunidades de aprendizaje, son esquivados, resueltos por el encargado de laboratorio o remitidos a técnicos expertos que se encargan de reparar las *netbooks*, cuyo lugar de trabajo está en Buenos Aires, de donde algún día volverían "reparadas". El docente se refugia tanto en la simplificación —tema tratado en el Corolario final— que termina perdiendo su espacio curricular para convertirse en un administrativo o personal *ad hoc* de profesores de otras asignaturas, del equipo directivo o del área de administración —asociado a la figura metafórica del lobo solitario o estepario del capítulo 3.

A pesar de los airados comentarios escuchados en los entrevistados sobre este tema, cabe indicar que si las versiones de *Windows* y de *Office* son cada vez *fáciles* de usar —y admiten que lo único que enseñan es eso— se entiende que los terminen reemplazando por una figura neutra, que opina en cuanto al formateo de los textos, que aporta unas cuantas fórmulas elementales e imparte alguna que

otra destreza, como la de copiar archivos o subirlos a *Internet*. Por comparación, es bastante más complicado explicar matemáticas, enseñar a escribir sin faltas gramaticales o enumerar eras geológicas que enseñar a "usar la *web*", como si un navegador no fuera bastante intuitivo. Por esta razón los docentes de informática devienen en ayudantes de los profesores de otras materias, y no al revés.

Respecto del programa de estudio obsoleto antes referido, sirvió como punto de partida para armar dos tipologías, dos escalas construidas para poder medir cuánta complejidad y de qué tipo es aquella que manejan. La misma comienza en el nivel de menor resistencia al cambio, donde se ubica a *Windows + Office* (el tipo de herramientas que dominan los usuarios)<sup>56</sup>. Siguiendo este criterio, se determinó con mayor precisión las zonas de fricción y el círculo entrópico de datos en el que queda situada la escuela cuando docentes y encargados de laboratorio no quieren (RC1) o no pueden (RC2) aportar soluciones, esto es, soluciones que reduzcan las situaciones recurrentes en el trabajo que realiza el personal administrativo y docente.

Respecto de la base utilizada para crear la escala mencionada se construyó siguiendo dos de los cuatro modelos que estipula Weaver (1948) y que destaca Reynoso (2006) como apropiados para construir grados de medición, según el cual no hace falta comparar usando los cuatro modelos y, debido a la cercanía conceptual al área de trabajo, se clasificó a la población de estudio solamente por los modelos I y III. Por cierto, se sugiere el modelo II para matemáticos, estadísticos, especialistas en ciencias naturales o investigadores que cuenten con mayores y variadas poblaciones de estudio que las aquí empleadas. A su vez, el modelo IV está referido a profesores de literatura y filosofía, grupos de estudio no considerados aquí.

56 Este tema se introduce en el Capítulo 3.1.8. El problema de la escuela entrevista como empresa, y se amplía en el Apéndice A: "Entre *softwares* comerciales y liberados".

Hay una zona de fricción entre lo que la sociedad exige a los informáticos y lo que ellos entregan. La deficiencia en cuanto a la forma que abordan la complejidad se ha revelado como un aspecto clave. En un grupo RC1 se pone en evidencia la resistencia al cambio. Nadie puede obligar a los miembros de este grupo, puesto que su deber primario es atender a los alumnos, y tratar los demás casos solo en términos hipocráticos. En segundo lugar pueden no ver el objeto de emprender tal acción, niegan la necesidad o, simplemente, dedican sus esfuerzos a otros temas. Por tal razón se realizaron varias propuestas para reconectar las partes en conflicto, en base a una adecuada dirección por parte de los directivos, así como de planes de carrera para los docentes, tales que redignifiquen su profesión.

Respecto de RC2, el segundo grupo, éste tiene serios problemas para trepar en la escala construida aquí, compuesta por niveles de complejidad. Para este grupo se prescribe una combinación de terapias basadas en técnicas para romper bloqueos, enmarcadas en el adecuado tratamiento de una teoría que se supone, traen incorporadas para resolver casos de complejidad: ni más ni menos que la Teoría General de Sistemas, que no se trata correctamente en las carreras de sistemas<sup>57</sup>, y que cual constituye un nuevo hallazgo. En el Capítulo 3, se menciona el posible abordaje del tema mediante Teoría de Catástrofes, para casos en los cuales los problemas de comunicación con los pares y la incapacidad de afrontar el cambio ponen en riesgo la carrera, cuando no la salud del sujeto de estudio.

Por lo tanto, la escala planteada sirvió además para determinar mejor el perfil que es más útil en la escuela, encuadrado en base a requerimientos de competencias mecánicas – sistémicas, con ciertas graduaciones hacia una y otra punta del espectro, tales que redundarían en mejores ubicaciones de las personas

57 El tema se trata en el Apéndice C: El pensamiento estratégico necesario en las carreras tecnológicas, y específicamente en C.2: Tratamiento terapéutico en docentes de tecnología para afrontar cambio y complejidad

dentro de la escuela. Con este planteo se busca resolver el problema tanto de la excesiva hiperespecialización (muy rara en el aula tanto como muy necesaria para resolver problemas de automatización en la escuela), como de la excesiva simplificación de partes a los alumnos (los programas de estudio obsoletos ya mencionados). Varias líneas de acción se desprenden de estos conceptos, tales como, por ejemplo, la subestimación de gran parte del alumnado con herramientas ofimáticas de usuario, pensadas no para educar, sino para facilitar el papelerío administrativo en las empresas, máquinas de escribir y calculadoras sofisticadas. Y la solución basada en profundizar en tecnología o en ciencias no aplica para todos los casos<sup>58</sup>.

De esta manera, se propone la conformación de cursados breves de tecnología, a modo de talleres de poca carga horaria, semejantes a los que en otros campos se brindan en las escuelas técnicas o de capacitación laboral. También se podrían armar actividades optativas por créditos sobre redes LAN, páginas web, seguridad informática y otros temas. El mismo proyecto Program.ar y algunos otros referidos a construcción de robots<sup>59</sup> trabajan sobre esta línea de acción. Los alumnos raramente necesitan de muchas horas para determinar si continuarán o no por sí solos la práctica de las técnicas recibidas. A menudo basta con conectar dos máquinas, configurar un *router* o armar algunos bucles en un lenguaje de programación para que el alumno pueda darse una idea general de la actitud a emplear en este rubro: paciencia e indagación. De esta forma el alumno lograría clarificar mejor la elección de sus vocaciones y carreras de estudios

58 El tema se trata con mayor profundidad en el "Apéndice D: ¿Resolver problemas fundamentales o de las empresas?".

59 Anónimo (s.f.). Programa "Todos a la Robotica". Recuperado de: <http://www.diariouno.com.ar/mendoza/derecho-renueva-su-plan-estudios-primera-vez-25-anos-la-uncuyo-20160521-n793212>, [consulta el 15 de agosto de 2016].

formales en el campo. Asimismo, la universidad contaría con el beneficio extra de una disminución de las tasas de deserción.

Otro beneficio de contar con la mencionada escala descansa en el logro de un encuadre del perfil docente actualmente en ejercicio y la determinación de las tareas donde se sienten más cómodos (aula, laboratorio o administración). Ahora bien, esta no es una cuestión baladí, ya que la primera impresión en los jóvenes es clave para impulsar o no sus futuros estudios. Hay docentes con capacidades mecánicas superlativas que estarían más aprovechados impartiendo clases específicas o automatizando tareas para la escuela, antes que enseñando solamente ofimática elemental. De la misma manera, docentes sistémicos estarían más cómodos proponiendo un uso sinérgico de las propias herramientas ofimáticas, tales como —previa capacitación— *Google Docs*, colaboración mediante *Wikis*, aprovechamiento del *crowdfunding* y de *clouds* gratuitos (caso de estudio: *Pokemon Go*), entre otros ejemplos más cercanos al tiempo futuro en el que vivirán los alumnos, que al tiempo pasado en que vivieron sus profesores<sup>60</sup>.

La idea central del trabajo no es quitar del medio a los docentes actuales, sino trabajar junto con ellos, a fin de potenciar sus posibilidades y ganar en salud laboral y excelencia de servicio. La capacitación es una herramienta indispensable para el cambio de los bucles negativos entrópicos. Esta declaración se aleja completamente de las capacitaciones actuales propuestas, por las cuales los docentes rapiñan alguno que otro punto para mejorar su bono, con conocimientos que insultan el nivel de lo aprendido en la universidad. Por tal motivo lo que se propone son planes de carreras mediante desarrollo de habilidades específicas. En todos los casos entrevistados, hubo una actitud entusiasta respecto de la propuesta de acceder a cursos de especialización, por la simple razón que estos

60 Se incluyen varias sugerencias en el Apéndice B: Propuestas de escenarios educativos para el docente de TIC.

cursos les permitirían llegar a las empresas o a la industria, sin que esto deba percibirse como una fuga de cerebros desde la escuela. Ello sí podría ser solventado por el Ministerio de Educación, y redireccionar fondos expurios hacia áreas de capacitación que reduzcan el malestar (que da lugar a enfermedades, mala atención, cumplimiento ineficiente de las tareas y pérdida de recursos educativos por mal uso de capacidades y potenciales humanos existentes). Esto no sería una pérdida de presupuesto: el sistema educativo no ha podido proveer hasta la fecha de recursos humanos capacitados, ni brindar capacitaciones que generen un cambio real en cuanto a la formación educativa en materia tecnológica, pese a la gran inversión y el énfasis de los programas implementados (1 a 1 y Conectar Igualdad).

Así pues la reubicación de los docentes con baja motivación que atestan las escuelas se daría de forma progresiva, natural y voluntaria. En las escuelas quedarían los que tienen mayor vocación docente, que afines y propiciadores de innovaciones o abiertos a la actualización tecnológica e informática, desarrollarían una identidad proactiva, tal que difícilmente pudiera ser subestimada, a quienes difícilmente se les asignarían tareas estadísticas de administración o secretaría "para que hagan algo"

Se torna indispensable pensar o, mejor aún, repensar el lugar de la informática, la tecnología y los profesionales asociados a este campo eminentemente educativo, que hoy por hoy no logra hacerse cargo de su quehacer ni de su rumbo o nicho sociocultural. Una reingeniería organizacional de las competencias profesionales desde una perspectiva sistémica —como indica Guy Le Boterf (2001)— resulta imprescindible, a fin de *aggionar* la formación de las nuevas generaciones, so pena de generar infradotados tecnológicos incapaces de adaptarse por vías educativas formales, a los avances irrefrenables de las nuevas tecnologías de comunicación y la información. Hoy es una realidad que la mayoría

de los educandos aprende el uso de las NTIC fuera de las clases de Tecnología o Informática. Difícilmente los estudiantes que transiten por el sistema educativo formal vigente puedan cultivar capacidades, destrezas y saberes en el más amplio sentido del término, que los posicionen significativamente frente a un futuro profesional laboral consciente y preclaro de sus horizontes y líneas de intervención, a la vez que de un acceso pleno al mundo del trabajo. Una amplia tarea y labor investigativa queda por delante, de las cuales el presente trabajo solo pretende ser un humilde escorzo.

## Apéndice A - Entre *softwares* comerciales y liberados

### A.1. Introducción

En la introducción al presente trabajo se había mencionado que el disparador a las preguntas de investigación fue si el *software* libre era una resistencia específica al cambio, o si era un síntoma de otras resistencias a las nuevas tecnologías. El *software* libre no es por supuesto una "nueva tecnología", puesto que su formulación teórica existe desde 1983 con la creación de la Licencia General Pública, y con su aplicación más emblemática, el sistema operativo *Linux*, presentado en 1989 en la Universidad de Helsinki. Su uso es común en el ambiente empresarial y científico. Sin embargo cobra la categoría de "nuevo" para los docentes que lo poseen en las *netbooks* y que lo encuentran en otras migraciones a *Linux* que algunos encargados de laboratorio han realizado espontáneamente en otras partes del país<sup>61</sup>.

Si bien en el análisis de datos se demostró que la resistencia al cambio es efectivamente, generalizada a cualquier dispositivo nuevo, el presente capítulo explora específicamente la resistencia del uso de *software* libre. Para ellos, revisa

61 Establecimientos que migraron por proyectos propios a *Linux* o a *Software Libre* antes del impulso de Huayra o de *Conectar Igualdad*. De acuerdo al censo de la comunidad de maestros *Gleducar*, y sumadas a las del proyecto *Lihuen*, suman 42 establecimientos entre escuelas, comedores y escuelas hogar.

- Anónimo (s.f.), *Escuelas que utilizan Software Libre*, [http://lihuen.info.unlp.edu.ar/index.php?title=Escuelas\\_que\\_utilizan\\_Software\\_Libre](http://lihuen.info.unlp.edu.ar/index.php?title=Escuelas_que_utilizan_Software_Libre), [fecha de consulta: 21 de agosto de 2016].
- El sitio mencionado se encuentra fuera de línea. Sin embargo puede obtenerse copias guardadas en el sitio *Archive*. Por tal razón la citación pasa a expresarse de esta forma: Gelbort, R (2013), *Escuelas usando Software Libre*, [https://web.archive.org/web/20130509200912/http://wiki.gleducar.org.ar/index.php/Escuelas\\_usando\\_Software\\_Libre](https://web.archive.org/web/20130509200912/http://wiki.gleducar.org.ar/index.php/Escuelas_usando_Software_Libre), [fecha de consulta: 21 de agosto de 2016].

la validez de diversos argumentos que sostienen tanto sus partidarios, como los que oponen de los usuarios de *software* cerrado.

Mientras que los segundos señalan que el *software* libre es innecesariamente complejo —así como otros argumentos que luego señalaremos—, los otros apuntan a que los usuarios de *software* cerrado se prestan a un juego de mercado pernicioso.

De esta manera, mientras que en los primeros capítulos se explicarán algunos conceptos necesarios, y se desarrollaran estos argumentos, en el resto del capítulo se intentará falsarlos, para dar lugar una discusión de tipo ética en el espacio áulico. Su residuo debería ser una comprensión cabal del hecho que ambas tecnologías (Internet mediante) se encuentran intrínsecamente unidas, y de que no se puede obviar el aprendizaje de una u otra sin caer en omisiones importantes.

## **A.2. La superestructura *Wintel* de los *software* comerciales**

El círculo *Wintel* es una expresión referida a un juego de mercado: un modelo de venta de *software* y *hardware* en el que todos los consumidores de tecnología están incluidos. El círculo *Wintel* es en cierta manera el impulsor de la famosa Ley de progresión de Moore, según la cual todo procesador dobla su velocidad cada dos años. Actualmente se puede remplazar la expresión de velocidad en Mhz a la adición de más núcleos de procesamiento paralelo, por lo cual la ley sigue vigente. Este círculo se explica de la siguiente manera:

- una empresa de venta de procesadores, para poder mantenerse sobre las otras, debe competir con modelos que las otras no alcancen a copiar a tiempo;
- para solventar la investigación de nuevos modelos, la empresa debe lograr una entrada permanente de dinero. Sin embargo, la cantidad de consumidores es limitada. El mercado no es infinito. Entonces, ¿cómo se hace para que personas que ya compraron un procesador, se deshagan de él y compren otro nuevo? Hay dos respuestas a este interrogante:
  - fabricar procesadores obsoletos. Así como las lámparas, las impresoras y la mayoría de los electrodomésticos que usamos, también los procesadores son defectuosos de fábrica. A esto se le llama *obsolescencia programada*<sup>62</sup>; y
  - realizar una venta cruzada de acciones entre las compañías que fabrican procesadores (con la compañía que fabrica el sistema operativo hegemónico usado en los escritorios hoy en día, *Windows*). La compañía que los sustenta emite sin razones verdaderamente valederas, versiones sucesivamente más pesadas en cuanto a requerimientos, que exigen lógicamente del último *hardware* disponible para correr. Cualquier usuario, con un poco de entendimiento de los tiempos de latencia en la respuesta de *Windows 2003 Server* o *Windows XP*, se dará cuenta que las ventajas de *Windows 2012 Server* o *Windows 10* pueden ser

62 La lectura de este artículo puede dar una idea más acabada de lo defectuosas que son las piezas que compramos a diario:

Ambit, M. (2012). *La bombilla que lleva más de 100 años encendida*. Obtenido en <http://www.esencialblog.es/la-bombilla-que-lleva-mas-de-100-anos-encendida/>, [fecha de consulta: 9 de marzo de 2014].

añadidas a las anteriores con un simple paquete de parches o *servipack*; y

- aproximadamente cada cinco años, los usuarios deben desechar sus equipos y adquirir otros nuevos, para poder tener el último *Windows* funcionando.

Solo como comparación los sistemas *Unix FreeBSD* y *Linux Debian* (producidos por entidades sin fines de lucro) sacan cada año versiones recientes (*releases*), que técnicamente pueden ser instalados en los viejos procesadores originales multitarea *Intel 80386*, salidos al mercado en el año 1985. Esto los hace mucho más democráticos, sociales, y energéticamente eficientes. Por lo cual se debería determinar que *Wintel* es malvado. Sin embargo esta acepción es falsa desde una perspectiva evolutiva —como veremos en el siguiente capítulo.

Pero antes, ¿es la solución recomendar la adopción de sistemas operativos libres para quebrar el círculo *Wintel*? Probablemente no. No se trata de generar una propuesta anárquica, pues el principio de la anarquía es contrario a cualquier teoría de sistemas lógicamente fundada, si tiene ésta pretensiones de consistencia interna. Para empezar, es casi imposible detener la inercia instalada por la macroestructura sistémica de *Windows*. Pero, además, resulta irrelevante, pues los usuarios "finales" o "de escritorio" son felices con *Windows*, y lo que hacen en realidad, es usar *Linux* por debajo cada vez que acceden a un router, al correo o a acciones tan simples como pagar una factura o guardar archivos en nubes (*Dropbox, Google Drive, Mega, etc*).

El minoritario *Linux* se constituye en una macroestructura sistémica que responde a otra lógica estructural. Una lógica que establece reglas inter e intraparadigmáticas de traducción y conversión respecto de sistemas como el de *Windows* y *Apple*. Aunque aparezca *Linux* como minoritario, apenas como

paradigma alternativo, minoritario y lateral, en realidad atraviesa su mejor momento, el cual si bien fue en un principio académico-científico, hoy en día es social. Esta adopción ocurrió en el primer decenio del siglo XXI, cuando se adoptó masivamente tanto en *clusters* de alta disponibilidad como de alto rendimiento (render, simulaciones de fluidos, etc).

Actualmente, en el segundo decenio, también se lo encuentra en casi todo sistema embebido: desde celulares a máquinas expendedoras de gaseosas. Los usuarios cuando usan *Windows*, no se dan cuenta que están usando una cascara, una *interfase* que accede a un mundo compuesto por *clusters*<sup>63</sup>. Sólo *por conectarse para hacer algo*, como bajar el correo (si es público) o pagar un servicio, se está accediendo potencialmente a una nube de "pingüinos"<sup>64</sup>. Incluso cuando se accede a páginas hogareñas, personales o de pequeñas PyMES, éstas están corriendo sobre alguna VPS (*Virtual Private Server*) económica virtualizadas por muy poco dinero, o están montadas en *hostings* compartidos, *desplegadas* por valores monetarios aún más económicos.

63 Alonso, S (2011). *A quién le importa Windows*. Recuperado de <http://bunker-blog.blogspot.com.ar/2011/01/quien-le-importa-windows.html>, [consulta el 21 de agosto de 2016].

64 Anónimo (s.f.). *Treemaps*. Recuperado de <http://www.top500.org/statistics/treemaps/>, [consulta el 21 de agosto de 2016]

### **A.3. Inducción a la obsolescencia programada: el dilema del gasto energético**

Wintel, con su ciclo de descarte continuo de tecnología, también puede examinarse desde el punto de vista del derroche. Se ofrece a continuación un argumento seguido de una falsación.

Pocas personas saben lo mucho que el consumo impulsa, a través de los impuestos y a través del abaratamiento de piezas, la investigación científica. El ciclo puede expresarse de la siguiente manera:

Ciencia → Tecnología → Consumo → Abundancia de dispositivos tecnológicos complejos → Derroche.

Hoy en día la CPU de cualquier teléfono celular es superior a los procesadores empleados en el Apolo 11 o en las sondas Voyager. Los científicos a cargo de los proyectos ya no necesitan crear procesadores nuevos. En todo caso necesitan proteger los actuales de la radiación. Con lo cual es llamativo el hecho que a través de la invención de partes y continuo desecho de las obsoletas, el hombre en una suerte de torpe evolución, sea potencialmente capaz de alcanzar otras estrellas. Figurativamente, puede alcanzar el universo edificando sobre en una montaña de basura tecnológica, y causando por el camino un impacto ecológico nefasto sobre la naturaleza. Temas que, nuevamente, la teoría de sistemas considera de diversas maneras.

Este estado de situación ha movido a los ecologistas a sostener que deberíamos cambiar nuestra actitud frente a la naturaleza antes de salir a contaminar el universo y de hacer de la luna un basurero cercano.

Sin embargo, *Wintel* es sólo una forma tecnológica de evolución darwiniana: todo componente que no puede ser más rápido queda obsoleto; y, aparentemente,

forma parte de la homeostasis de las empresas de tecnología. La idea de ser conservador con el *hardware* es falsa desde el momento en que a nuestra velocidad actual de contaminación, no hay manera de que podamos contaminar el universo sin antes presenciar su muerte energética, o nuestra propia muerte como especie. El argumento de los ecologistas solo se mantiene limitado conceptualmente a la nave "tierra", no a nivel universal. Por eso quizás este capítulo sea tan irrelevante como la ecología misma que usa como defensa, y debido a esta razón no se la ha propuesto como hipótesis. Su utilidad es apenas una práctica de falsacionismo.

La única utilidad que posee el argumento, acaso el residuo cognitivo que deja su reflexión, es la discusión que propone respecto de la cantidad de recursos naturales que necesita tomar *Wintel* para mantenerse en estado de homeostasis negentrópico. Esto es, cuanto prestarse al desecho continuo de piezas y al crecimiento de los cementerios de tecnología<sup>65</sup>. La manera de ralentizar el círculo es atacando sus arbotantes: consumir menos. El problema para lograr esta ralentización es que quien pretenda vivir sin consumo innecesario, incluso cuidando su máquina *defectuosa de fábrica*, tendrá que retornar al círculo tarde o temprano, ya que eventualmente no conseguirá piezas de recambio.

### **A.3.1. Debates contemporáneos en torno al lugar y el impacto de las nuevas tecnologías**

*Wintel* genera una compra de componentes y dispositivos con obsolescencias programadas, cuyo impacto ecológico resulta perverso. No obstante es un factor de desarrollo tecnológico. Cada componente como así también cada dispositivo, ni bien surge busca ser superado cualitativa y

<sup>65</sup> La expresión refiere al problema que suponen ciertas aleaciones complejas, altamente contaminantes y muy integradas, que resultan imposibles de reciclar.

cuantitativamente por nuevos diseños más veloces. Obsolescencia + Innovación = desarrollo tecnocientífico.

La innovación requiere de financiamiento que está asociado al consumo de productos tecnológicos. Por lo tanto, a mayor obsolescencia, mayor retorno financiero de divisas que posibilitan la reinversión en procesos de innovación tecnocientífica. Esa tasa de retorno busca tener circuitos de realimentación cortos y con desarrollos geométricos para que la tasa de inversión futura sea mayor que la inicial de inversión. El consumo de las masas apoya indirectamente la investigación, y por lo tanto, el desarrollo tecnológico. Poca gente sabe lo mucho que ha bajado el presupuesto espacial desde la llegada del hombre a la Luna. Incluso la NASA se maneja con presupuestos paupérrimos en comparación a las misiones *Apolo*. Actualmente si podemos darnos el lujo de seguir mandando sondas al sistema solar, es porque muchas de sus partes son *outsourced*: chinas, hindúes, incluso argentinas. Circuitos comunes y corrientes protegidos contra radiación: material de investigación descartable.

Hasta el momento no existen dispositivos tecnológicos electrónicos de contaminación cero, donde tal ideal sólo es sustentable por las culturas paleolíticas, las cuales resultan incompatibles con el *modus vivendi* de las culturas tecnoconsumidoras del siglo XXI. Pareciera que no podemos evolucionar y expandir nuestro espacio vital sin desordenar y desequilibrarlo un poco. El problema es que este "poco" suele tener fuertes consecuencias de impacto ambiental. Las progresiones de consumo son miles de veces mayores a la progresión de reasimilación y reincorporación de los ecosistemas naturales de donde proceden los insumos empleados, los cuales son muy difíciles de reconstituir o son directamente irrecuperables. Incluso habiendo muchos cientos de millones de seres humanos viviendo del reciclado y de la separación de partes, haría falta muchos otros cientos de millones para reciclar el total de lo desechado.

Por supuesto predicar la necesidad de que haya más gente pobre para que haya más reciclaje es un locura. De esta manera al parecer la humanidad propia se encuentra en un círculo al parecer imposible de romper. Quizás la especie se encuentre en un estado kármico donde la propia naturaleza de las cosas se encuentre condenada. Es la razón por la cual Watzlawick (1995, pp. 22) en "El sinsentido del sentido" interprete la inquietud de Max Planck respecto de la voluntad, "la cual contemplada desde fuera está determinada causalmente, y que contemplada desde adentro, es libre".

Marcelo Pakman lo expresa con mayor claridad cuando introduce la obra de Foerster (2006, pp. 33)

"[...] el núcleo del trabajo es un intento de caracterizar a los sistemas capaces de organizarse a sí mismos, de modo que no sea contradictorio con la segunda ley de la termodinámica. Esta ley establece que en un sistema cerrado a los intercambios materiales y energéticos con su ambiente, hay una magnitud llamada "entropía" que debe aumentar en forma constante hasta llegar a un máximo. Dicha magnitud es una medida de desorden del sistema [negentropía], de la incapacidad para transformar energía en trabajo, y puede entenderse como la inversa de la información porque a más desorden, más incapacidad de un observador para decir algo (informar) sobre un sistema. La equivalencia entre la entropía y la inversa de la información hace del concepto de "entropía" la noción puente entre el mundo de la física (mundo de la materia y la energía) y el mundo de la cibernética (mundo de la información y de la organización). De hecho aquellos que hablan del nivel organizativo, sin incorporar el lenguaje de la cibernética, lo hacen en términos de entropía (por ejemplo Prigogine)".

Especialmente es aquí donde Pakman subraya la idea:

"Particularmente fértil resulta la noción de que todo fenómeno de auto organización va acompañado de uno de desorganización en el ambiente

("exterior" o "interior") del sistema auto organizador. Si se trata de eliminar dicho aspecto desorganizador, también perdemos el fenómeno auto organizador, consecuencia de ello es que todo sistema de este tipo requiere de un ambiente del cual tomar orden y al cual desorganizar. Para poder organizarlo, ese ambiente debe tener cierto orden, caso contrario no habría que desorganizar. Apasionantes problemas éticos presenta esta formulación cuando lleva al campo de lo social. ¿Cuáles son los límites y reguladores que debemos incorporar en nuestra práctica cuando pretendemos incrementar la organización de un sistema familiar o terapéutico, si nos hacemos cargo de que, por necesidad, la desorganización acompañará en el proceso en algún otro dominio?".

Estas palabras se ven reflejadas en una entrevista en 2012 con el Ing. Antonio Gálvez, de Siemens Chile, dando una opinión tan cruda como vigente: "en ninguno de los congresos sobre energía a los que he asistido, se vislumbra a corto plazo, ni a 140 años en adelante, que las energías alternativas, o que las acciones ecológicas de los gobiernos, puedan abastecer al mercado. Las personas quieren calefacción, refrigeración, confort. Quieren energía barata, heladeras, conducir a su trabajo, y no están dispuestas por ninguna razón a renunciar a todo eso". Por lo tanto, probablemente no haya interesados en frenar la inercia del círculo *Wintel*. Incluso si las computadoras actuales funcionaran a vapor, y fuera *Windows* su carbón, y todo el conjunto apestará por doquier, nada de eso se dejaría de usar, ya que el conjunto ha pasado de ser "el medio para lograr algo", a un "todo con el usuario: una extensión del sistema límbico". A esto se refiere Ashby (1960, pp. 87) cuando menciona que "*el organismo afecta al medio, y este al organismo; de un sistema de este tipo se dice que hay retroacción o realimentación*"<sup>66</sup>.

66 En inglés, el original: "*The organism affects the environment, and the environment affects the organism: such a system is said to have 1 feedback*".

De este modo, los problemas de la tecnología no se pueden resolver inhibiendo su presencia, como pregonan diversas sectas ortodoxas (neofóbicas). Paradójicamente, los problemas de la tecnología sólo se pueden resolver con más tecnología: investigación en las tecnociencias orientadas hacia desarrollos tecnosustentables y sostenibles.

Esta tecnología en su diseño debe incluir una recolección del desorden que provoca, o ser descartada de plano. Ello implica inversiones más fuertes, donde junto la invención de un objeto se incluya la previsión del impacto entrópico sobre el sistema planetario y los seres vivos que lo habitan. Todo lo contrario es generar un producto "defectuoso de fábrica". Vender un sistema operativo que requiere de desechar todo lo anterior, es ineficiente (no sustentable ni sostenible), desleal (porque se basa en leyes mercantilistas irresponsables), inmoral (por lo perverso) y antiético (por su impacto en la vida y las condiciones a las que expone)<sup>67</sup>.

67 Reynoso (2006, pp. 134) "Cuando se quema un combustible no hay manera de recombinar la energía perdida en el calor atmosférico y volver a obtener lo que se tenía antes. La materia y la energía no son intercambiables, pues entre ambas se interpone cierta degradación o entropía. La concepción del universo surgida a la luz de la termodinámica se refiere a un agregado masivo de formas de energía que paulatinamente se están agotando, en un camino irreversible hacia el colapso final. Pero no es posible (dice Adams) que una ley tan lúgubre sirviera para explicar eventos expansivos, como parecía ser la civilización tras la revolución industrial. A principios del siglo XX la teoría darwiniana de la evolución se unió a la segunda ley de la termodinámica para explicar por qué en lugar de seguir un proceso de degeneración caótica las especies vivientes tienden hacia formas de existencia cada vez más organizadas, expansivas y complejas. Alfred Lotka (quien formuló inicialmente estas teorías) sostuvo que esta aparente violación de la segunda ley de la termodinámica se debía a que las especies utilizaban cantidades cada vez mayores de energía, extrayéndola del medio ambiente. Las especies que gastan mayores cantidades de energía poseen ventaja sobre las que consumen menos, las que en general tienden a convertirse en sus presas. Algunos antropólogos de las décadas de 1940 y 1950 congeniaron con estas ideas, como fue el caso de Fred Cottrell y sobre todo Leslie White. Después de la Segunda Guerra se comprendió que no son las especies las que

Intel ha hecho algunos avances en los últimos años en tal sentido, permitiendo que procesadores más potentes ocupen *sockets* estandarizados de *motherboard*<sup>68</sup>. Es de esperar que mantenga esta filosofía y que sea copiada por otras compañías que forman parte del círculo, como AMD y VIA. Sin embargo, el *software* comercial debe mejorar en eficiencia y no en efectos especiales; un gran cúmulo de excepciones simulan una pseudoestabilidad, una constelación de excepciones, que cubren una deficiente arquitectura monolítica, con más bucles que atrapan errores, que bucles afines a sus fines.

#### **A.4. Prejuicios verdaderos y prejuicios falsos**

Entre las observaciones que dieron origen a las primeras preguntas de investigación del presente trabajo, se barajó la posibilidad que la resistencia durante la mencionada migración a *software* libre fuera solo un prejuicio. Tras el análisis de datos realizados, ahora sabemos que forma parte de un conjunto más amplio de resistencias y problemas de acceso a la complejidad.

Sin embargo, no se termina de eliminar del todo la pregunta original, por cuanto entre los problemas de acceso al cambio se encuentra varios sesgos cognitivos —varios de ellos pueden observarse en las entrevistas—

constituyen las unidades de supervivencia, sino los sistemas de especies en el contexto de su medio ambiente. Esta es la clave del modelo ecosistémico, elaborado en ecología humana y animal por Howard Odum, Ramón Margalef y otros autores".

68 Concretamente, el *socket* LGA 1155 (Sandy Bridge). Técnicamente, se puede instalar allí un procesador i3, y en caso de requerir más potencia, cambiarlo por un i5 e, incluso, un i7, sin cambiar todo el *motherboard*.

Entre los sesgos detectados, se encuentra evidencia respecto de prejuicios en relación con el *software* libre, específicamente entre quienes se han "formado" bajo la matriz del círculo *Wintel*. Se examinan estos prejuicios con el fin de rodear de un marco aquellas líneas de acción que quiera seguir el lector en aras de demostrar aquellas que le parezcan fructíferas.

#### **A.4.1. Gratuidad y la calidad del *software***

Un prejuicio respecto del *software* es que al ser gratis y proveniente de una compañía desconocida, no ha de ser tampoco de buena calidad. A ello se suma que muchos programas en Internet son gratuitos y, generalmente, son nocivos (auténticos caballos de Troya)<sup>69</sup>.

Entonces, ¿es "malvado" el *software* gratis? El gran problema aquí es que el docente de tecnología no entiende o no establece la diferencia entre estos *freeware* (*software* gratis) y aquellos *opensource* (código abierto). La diferencia entre ambos es fundamental. En el caso de los programas *opensource*, se dispone de la fuente para auditar, en tanto que en el *freeware* sólo cuenta con un misterioso binario ya compilado. O como dijera el arquitecto de la película *Matrix*, "*the door to your rights leads to the Source, and the salvation of Zion*". Debido al sutil juego de palabras en inglés, lo traduzco arbitrariamente, como "*la puerta a tu derecha —o a tus derechos (rights)—, te llevará a la fuente y la salvación de Zion*".

69 *Trojan Downloaders* (bajadores de virus) no son detectados por el antivirus, porque técnicamente no son virus. Una vez instalados, mediante DDNS, estos programas realizan una conexión inversa encriptada hacia un *host* cuya ip es dinámica, que baja el auténtico código vírico.

#### A.4.2. Totalidad versus Modularidad

Otra de las razones que provoca rechazo en los programas de código abierto es su sensación de incompletitud, respecto de los dispositivos gigantes multipropósito comerciales. En este caso, lo que no se entiende es la *modularidad* del *software* libre, la regla de escribir "un programa que haga una sola cosa, pero que la haga bien" (Salus, 1994, pp. 52, citando un comentario de Doug McIlroy). Contemplado de esta manera, los programas abiertos no son incompletos, sino que son pequeños —lo cual es distinto—, puesto que están pensados para comunicarse con otros programas (no con personas), hasta formar un todo. De acuerdo a la cita textual de McIlroy y Pinson (1978, pp. 1902):

*"Escribe programas que hagan una sola cosa, y que lo hagan bien. Escribe programas que puedan trabajar con otros programas. Escribe programas que puedan manejar cadenas de texto, porque esa es la interfase universal".<sup>70</sup>*

Estas partes construyen el *todo*, y se conforman en la libertad que tiene el usuario para decidir el producto final, sin que por ella deba saber programar. Se trata de una libertad de construir el sistema y no que el sistema venga impuesto por un suprasistema comando (la alegoría de la *matrix*). Tal situación se encuentra dada por la percepción holística de los sistemas informáticos cerrados y enlatados, concebidos como irreductibles, cuya apertura equivale a la pérdida potencial de datos: la caja de Pandora por definición. El docente trata como si fuera un usuario a los sistemas; los trata como impredecibles *gizmos*, cajas negras sobre los cuales

70 El párrafo citado se encuentra arbitrariamente traducido del texto original que reza: *"Write programs that do one thing and do it well. Write programs to work together. Write programs to handle text streams, because that is a universal interfase"*.

no se puede ni se debe<sup>71</sup> aventurar conjetura alguna respecto de su funcionamiento interno.

#### **A.4.3. Protointerfaces resistentes *ex professo***

Este prejuicio es verdadero y tiene bases sólidas. Usar *software* libre es más difícil. Esto se debe a los programadores de *software* libre normalmente no se preocupan por hacerlo "fácil" de usar, ni siquiera en un aspecto comercial. En todo caso los esfuerzos extras se hacen en función de adecuarse a estándares de la industria, o intentando entender diccionarios cerrados en aras de lograr la compatibilidad con productos cerrados<sup>72</sup>. Ese tiempo lo recuperan ignorando estándares de usabilidad. Otras razones pueden enumerarse de la siguiente manera:

- a) *practicidad*: "todo buen trabajo que empieza un programador, lo hace para rascarse una comezón personal" (Raymond, 2000, pp. 23). Se trata de automatizar un problema recurrente, de tal manera que no vuelva a molestar. Este detalle por sí mismo debería ser todo un aspecto de calidad;

71 De hecho, la ingeniería inversa sobre los sistemas cerrados se encuentra penada por leyes y tratados en países sujetos al ALCA, tratado al cual Argentina rechazó pertenecer el 5 de noviembre de 2005, durante la IV Cumbre de las Américas en Mar del Plata.

72 El esfuerzo de adecuación a estándares es mayor, por cierto, que el que hacen las empresas de software cerrado. A modo de ejemplo:

- Microsoft ActiveX, solo compatible con Microsoft Explorer, respecto de Javascript, compatible con cualquier navegador.
- Adobe Flash, propietario respecto de HTML5, el cual es el estándar dictado por el consorcio W3.

- b) *relación sincrónica*: pasado un nivel de complejidad, el programador pretende que los usuarios estén a su mismo nivel (no le interesa los neófitos) en "su" campo. No brinda una mesa de ayuda para los novatos, incluso aunque le paguen. Espera, en cambio, rodearse de un conjunto de entusiastas que hagan este trabajo. Esa es la razón por la cual regala su *software*, para que otros le ayuden a mantenerlo, y si tienen ganas, difundirlo o vivir de él. También espera poder librarse del *software* una vez que haga lo que necesita, sin ser molestado todo el tiempo por mejoras o versiones nuevas; y
- c) ausencia explícita de *interfases amables*: esto se desprende de los rasgos anteriores. Esta actitud segregativa proviene propiamente del programador, no del usuario. Inconscientemente los programadores de *software* libre siguen algunas recomendaciones de Jakob Nielsen sobre usabilidad para *interfases* Web. Específicamente, el tipo de mercado buscado puede suponer no solo un desperdicio de esfuerzo en hacer más atractiva la *interfase* a los "torpes", sino que en ocasiones hasta puede redundar en un perjuicio, ya que luego hay que atenderlos y complacerlos en sus necesidades particulares. Es más probable que un experto se detenga a leer un largo texto sin formatear, antes que leer una página web dotada de un diseño recargado que lo "ahuyente" incluso antes de entender de qué se trata. Allí el programador incluso se arriesga a que su *mercado*, a la manera de los consumidores expertos, desconfíe de encontrarse un *catch* —o venta solapada— por detrás<sup>73</sup>; y

73 Esta es la razón por la que *Google* evita usar exceso de elementos visuales en sus publicidades *adwords*. Algunos sistemas operativos de base *Unix* como *FreeBSD* (usado en *Google*), *Debian* (usado en servidores) o *Gentoo* (Instituto Balseiro) son muy buenos ejemplos al respecto, sus comunidades no mejoran las *interfases* simplemente para filtrar a los novatos. A menudo hacen bromas respecto de los esfuerzos de comunidades como *Ubuntu* o *Linuxmint*

d) la *intermicroconectividad programática*, es aquella propiedad en virtud de la cual los programadores de *software* libre generalmente escriben programas *cuyas salidas sean la entrada de otros programas*, no un usuario. Esto es una herencia del mundo de *Unix*, la "Regla de Composición" descrita por Raymond (2004, pp. 15) de la siguiente manera:

"Unix tradition strongly encourages writing programs that read and write simple, textual, stream-oriented, device-independent formats. Under classic Unix, as many programs as possible are written as simple filters, which take a simple text stream on input and process it into another simple text stream on output".

De esta manera lo que se traslada es la discusión a otro nivel. En este caso sería: ¿sirven las *interfases* textuales para aprender? Si bien no es el propósito del presente documento explotar esta idea, se deja los siguientes elementos para aquellos que deseen continuar el tema:

1. hay tantas personas que se sienten cómodas trabajando con interfases textuales, como aquellas que necesitan de *interfases* gráficas amables para poder desempeñarse. Por ejemplo esta tesis fue escrita usando *Openoffice*. Sin embargo se consideró varias veces crearla en *SGML* o *Latex* y compilarla usando *Openjade*, al estilo de los libros de O'Reilly; y
2. es muy fácil crearle *interfases* amables a los programas libres: solo se debe "hablar con ellos" en el lenguaje que esperan recibir: simple texto (ni siquiera binario como el *software* cerrado). Se sugiere al docente la investigación de librerías como *wxPython*, o *Shoes* para corroborar la validez de lo aquí expuesto.

por interesar a las masas. En este caso existe un pasado elitista muy difícil de revertir.

Sergio A. Alonso - 231 - Resistencia al cambio en torno a las NTIC

#### **A.4.4. El condicionamiento de las *interfases visuales***

Antes de desarrollar este punto, y refiriendo específicamente al tema de las interfases gráficas, se extrae el pensamiento de Eduardo Escalante Gómez (2002, pp. 153), quien manifiesta una cierta desconfianza respecto de las cómodas *interfases* visuales, si es que éstas solo son usadas para saltar cómoda e insaciablemente en busca de novedades banales, la cual "tiende a obturar la capacidad de reflexión y de análisis crítico de la persona".

Gómez, quien por su profesión indudablemente tiene una posición más imparcial que la mía respecto del *software*, apunta también por debajo de la línea de flotación a las *interfases enriquecidas* basadas en ventanas:

"La literatura señala que la escritura promueve la abstracción, el pensamiento lineal y la atención a lo inteligible; lo visual promovería, por el contrario, lo sensible, lo inmediato y, por ende, el debilitamiento de nuestras capacidades conceptuales. Entonces se podría construir la hipótesis de que la inteligencia interconectada corre el riesgo de disminuir el entendimiento abstracto, el entendimiento conceptual.

Actualmente, hay una inquietud respecto del impacto en la cognición del vértigo de la información en ambientes Windows. Preocupa que va a pasar con las habilidades que incluyen el estímulo a la reflexión, al pensamiento crítico y el respeto al multiculturalismo<sup>74</sup>.

Si partimos del conocimiento respecto que el cerebro se maneja en términos de economía, sabemos que éste preferirá las interfases visuales. Sabemos también que siguiendo el mismo principio, tienden a basarse en lo que ya conoce. Por lo tanto estamos en condiciones de señalar que las *interfases* visuales son

74 Con multiculturalismo nos referimos a la cantidad de idiomas al que es traducido Windows. En comparación, GNU/Linux, posee traducciones a lenguas cuyas economías no son representativas, como el Swahili, Aymara e incluso Esperanto.

condicionantes si son las primeras interfases a las que expone una persona, y que un cambio en la manera en que se hacen las cosas trastoca la manera en que esperamos que responda la computadora. Esta es la razón por la cual las empresas desarrolladoras de aplicaciones ponen especial cuidado en el momento de cambiar las *interfases*, para no perder usuarios.

Bajo *software* libre hay dos problemas relacionados con este hecho: por un lado, la ya mencionada *interfase*, que es prácticamente lo último que se programa, con poca o ninguna asistencia de un diseñador. El otro problema es que existen riesgos legales ligados a la propiedad intelectual que se pueden argüir al dotar a un programa de una interfaz semejante a una comercial. Muchas innovaciones visuales poseen restricciones de uso. Como ejemplo, legalmente no se puede imitar el *ribbon* de los nuevos *Office*. Sin ir a un extremo, en Estados Unidos de Norteamérica y en todos los países incluidos en el ALCA, los programadores deben además atenerse a patentes denigrantes como la presencia los botones o de los menús descolgables<sup>75</sup>. Técnicamente, el programador debería pagar por agregar funciones de doble *click*<sup>76</sup>, por usar la expresión ".*net*", o por incluir el *plugin* de *Flash* en el servidor de descarga<sup>77</sup>.

75 Como ejemplo, *Lotus* inventó en los años 80 para su hoja de cálculo "123", el menú superior descolgable. Sin embargo Microsoft obtuvo primero la patente.

76 Esta patente se encuentra registrada por Microsoft Corporation. De acuerdo al formato APA para patentes:

Lui, Charlton E., and Jeffrey R. Blum. United States Patent: 6727830 - Time based hardware button for application launch. 6727830, issued April 27, 2004. <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahhtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=6,727,830.PN.&OS=PN/6,727,830&RS=PN/6,727,830>.

77 Para no tener mala prensa, las empresas usualmente no accionan estas patentes hasta que sufren algún revés en la bolsa de valores, como el caso *Unisys* cuando comenzó sus demandas por el uso de los archivos "GIF" en las páginas web, o como el caso de Unix SCO,

Esta vileza genera una natural resistencia entre los programadores, quienes suelen generar alternativas libres, en lugar de archivos *gif* sugieren usar *png*; en lugar de *zip*, recomiendan comprimir con *tgz*; en lugar de *mp3*, enaltecen el uso de *ogg*, y así sucesivamente.

En conclusión, quien pierde es el usuario, atrapado en un juego de mercado que no entiende o que entiende pero que no domina. Aquí es donde el docente debería aclarar el tema.

que intentó demandar a *todos* los usuarios de Linux del mundo, por unas pocas contribuciones al *kernel*.

## Apéndice B: Propuestas de escenarios educativos para el docente de TIC

### B.1. Introducción

Cuando la Web llegó al aula, aproximadamente hacia 1998, los primeros docentes progresistas propusieron los *blogs* públicos como el arquetipo de tarea tecnológica; típicamente diarios escolares o investigaciones de los alumnos.

Y allí se quedaron. En ninguna de las entrevistas se encontraron tareas realizadas con diseños con lenguaje hipertextual, ni programación de ningún tipo.

Si en cambio, se manifestó la importancia de que las escuelas tuvieran Internet; los alumnos debían aprender a buscar y a encontrar. Pero mientras que el mundo literalmente explotó de sitios web, apenas hay referencias de crear en clase al menos un simple código al estilo

```
<html>
```

```
  Hola Mundo.
```

```
</html>
```

Como mucho los docentes progresistas tratan de dejar bien sentado que *no se oponen a los avances de las redes sociales*, y que en un dechado de modernidad suben los apuntes a *Facebook*. No explican muy bien como logran que los alumnos se concentran en los apuntes mientras la *interfase* los tienta con toda clase de chismes, juegos y propuestas de toda clase. Algo así como tratar de poner a un estudiante a repasar la lección en una playa o en un *shopping*.

¿Se ha llegado a un punto en el cual las redes sociales son la representación de la colmena humana, y no queda más espacio para la innovación? Las propuestas que aquí se presentan buscan llamar la atención respecto de los muchos proyectos áulicos que pueden marcar la diferencia entre alumnos consumidores de tecnología y alumnos creadores de contenidos, catalizadores de ciencia, verdaderos habitantes de un futuro que para sorpresa de muchos, ya llegó. Los estándares abiertos, el *software* libre, el pensamiento *hacker* bien pueden ser la clave.

## **B.2. El *hacking* bien entendido**

En los últimos años ha habido un movimiento de rescate y aclaración del término *hacker* por parte de aquellos que gustan de profesar la deconstrucción de los componentes *software* y *hardware*<sup>78</sup>. En el ambiente, se considera *hacker* al *especialista*: cualquier persona entusiasta sobre un tema, que al cabo de un tiempo de estudio se especializa al nivel de poder trascenderlo completamente y desarmarlo, incluso si este tuviera barreras para su estudio, barreras tales como ofuscación (complejización) o trabas legales. Esta es la razón por la cual el *software* libre, rechazado por los docentes de Tecnología, es en cambio muy bienvenido tanto en las comunidades *hacker*, quienes mediante la utópica expresión *hack the world* sostienen que los contenidos deberían estar abiertos para todos. Por la misma razón el *software* libre tiene buena recepción en las comunidades científicas. Los científicos son un buen ejemplo de *hackers*: cuando tratan de descomponer un átomo o un pulsar, lo que hacen es *hackear* los

78 Ciarlante, J (2016). *TEDxPlazadeMulas - Con permiso para hackear*. Recuperado de <http://tedxtalks.ted.com/video.mason/TEDxPlazadeMulas-Juanjo-Ciarlan>, [consulta el 25 de agosto de 2016]

elementos mismos de la existencia. Si hay información previa sobre esos elementos, tanto mejor, porque menos será la cantidad de *ingeniería inversa* que deben hacer.

Es muy fácil comprobar la polarización que hay de expertos informáticos y de científicos. Por alguna razón todos ellos se encuentran en los inquietos canales de charla IRC que se encuentran en la red de servidores *Freenode*.

Hay tres razones por las cuales estas dos comunidades se encuentran: 1) porque se halla reflejada en la práctica habitual que tienen los programadores de *software* libre de poner a disposición los códigos fuentes (no ofuscados) para su estudio y eventual mejora; 2) porque tiene que ver con la comunión que hay de *hackers* en torno a sistemas complejos, y refiere a la necesidad de ayuda que claman nuestros semejantes para entender el conjunto de partes, como la materialización de expresiones utópicas en el sentido de la *aldea global*, la *mente colmena*, la *sociedad del conocimiento* y otras expresiones equivalentes; y 3) porque se basa en el reconocimiento. La comunidad *hacker* otorga un valor mayor a aquellas personas que son capaces de ayudar a los demás a llegar a la comprensión. Nadie se auto adjudica el título de *hacker*: son los demás *hackers* quienes otorgan este "título" en base a la competencia evidenciada.

Formar a los alumnos con *tecnología software y hardware que puedan desarmar*, debería ser una práctica habitual en el espacio curricular de Tecnología; auspiciar la producción de tecnología en lugar de su mero uso. Ello constituye el primer escalón para un país que necesita salir al mundo con innovación en lugar de consumo. Incluso si no se puede llegar con las controvertidas *netbooks* a todas partes, máxime con el exigido presupuesto destinado a educación, bien se pueden dotar a los alumnos por unos pocos dolares —en relación a la cantidad adquirida— plaquetas Raspberry o Arduino munidas de Linux Ubuntu Mate, y conexión a periféricos en el aula de informática. El deseo por armar su máquina

debería ser un buen aliciente para el alumno, y un interesante experimento constructivista. Luego se volverá a este tema desde una perspectiva específica de los componentes físicos, en el Apéndice B.6. *Hardware Libre*

### **B.3. Bienvenidos al tren**

En el discurso oficial en boga desde hace unos años, es frecuente escuchar como desde el gobierno, denostan respecto del *establishment* y la colonización de lo que somos objeto los países emergentes por parte de los gobiernos de los países del primer mundo. Esta acusación, si bien es legítima, también ha sido usada a lo largo de la historia para distraer la atención de los votantes hacia un enemigo en común y polarizarla hacia metas conjuntas. Sin embargo, aquellos gobiernos imputados como "aprovechadores", no impiden que sus propios ciudadanos compartan ciencia y tecnología a los países emergentes, e incluso que hagan negocios *off shore* que ciertamente no benefician al fisco.

Es difícil ilustrar de una manera gráfica el gigantesco volumen de negocios que pierden los países desarrollados en el proceso de intercambio de tecnología y comunicaciones, a la vez que también es imposible medir en qué grado las creaciones de sus ciudadanos son mejoradas una vez que son compartidas. Para muestra basta con mencionar dos ejemplos: el *outsourcing* y el *software libre*.

El *outsourcing*, por ejemplo, es el fenómeno por el cual las empresas contratan recursos humanos de otros países. Muchos argentinos, programadores y diseñadores, tienen un buen presente a pesar de ser contratados por un sexto de su valor equivalente en los países desarrollados. Puesto que esta modalidad de contratación incrementa la desocupación en estos países desarrollados, el *outsourcing* podría ser fácilmente prohibido, simplemente bloqueando las cuentas

bancarias a los portales *freelancers*. Sin embargo hasta la NASA se le permite ahorrar costos de esta manera.

Por otro lado, el *software* libre también es un ejemplo de talento que se exporta a través de Internet: son potenciales productos sobre los cuales que no solo *no se fija copyright* (derechos de copia), sino que por el contrario, se los patenta explícitamente bajo licencias *copyleft* (garantía de libre copia).

Tan solo las donaciones a la humanidad de la red Internet (Estados Unidos) y por el otro de la Web (Suiza, Inglaterra), o al menos de su ausencia de bloqueo, son signos de que las sociedades del conocimiento son mundos posibles. Realidades que modelan y mejoran las realidades de aquellos que se convierten en ciudadanos constructores en lugar de inquilinos que solo se benefician de su indirecto confort. En tal sentido, la humanidad sería significativamente más pobre si el primer octeto de las direcciones de IP hubiera sido repartido solo a los países desarrollados, y sus datagramas ofuscados, mediante IPSEC (una encriptación extremadamente difícil de romper).

#### **B.4. La *netbook* como laboratorio**

Desde una perspectiva educativa, se debe llamar la atención sobre la conveniencia en el aula de usar *software creado en universidades*, en lugar de *software* comercial. Linux es conveniente desde el momento en que se encuentra inspirado en la visión original del profesor Andrew Tanenbaum, creador del sistema operativo Minix en la Universidad de Amsterdam, para poder explicar a sus estudiantes la naturaleza de los procesos internos de la computadora. Sin llegar al estudio de los sistemas operativos en si mismos, como fue su propósito original, Linux es una pieza de software que permite realizar varios experimentos

interesantes en clase, tales que llevan a la reflexión del alumno sobre varias hechos en particular.

El primero de los experimentos sirve para poner como pregunta de investigación la consabida queja respecto de la lentitud de las *netbooks*. Este es el consabido estigma de las *netbooks* distribuidas de por el programa *Conectar Igualdad*. ¿Es cierta esta aseveración?

Analicemos la computadora que usan los estudiantes. Tiene un procesador Atom de 1.6 Ghz, y 2 GB de RAM, una partición con Windows, y otra con Linux (Ubuntu, Debian, Huayra o LinuxMint). Es capaz de procesar 1.600.000 instrucciones en un segundo. En comparación y a grandes rasgos, la computadora de la misión Apolo 11 tenía 2048 Mhz y 2 KP (Kilo palabras de 16 bits). Es decir, la computadora de un alumno es 780 veces más rápida que la que llevo al hombre a la Luna.

Indudablemente, hay límites. Una *netbook* no es un *playstation*. El primer síntoma son la cantidad de objetos DOM que se alojan en el motor *javascript*, razón por la cual un exceso de banalidades en la línea de tiempo de Facebook tenderá a poner lento el navegador. Tampoco puede correr juegos con mucha carga gráfica.

Ahora que sabemos que es *lo que no puede hacer* una *laptop*, habrá que concentrarse en lo que *sí puede hacer*: conducir un módulo lunar hasta la Luna; ayudar a predecir un tifón; contribuir en la búsqueda de genomas defectuosos, controlar el Mars Curiosity Rover; ayudar a escribir una obra maestra y muchas otras tareas por el estilo. De lo que se puede deducir que *el límite no está en la computadora, sino en la escuela*.

¿Se puede ampliar sus límites? Técnicamente hablando, si se trata de aprender, nunca estuvo limitada. Las personas que se sienten "apretadas" por sus recursos, deberían entender que absolutamente cualquier tarea de aprendizaje con un programa pesado puede ser realizada con muchos programas equivalentes más livianos, pequeños, y que enseñan las mismas cosas. Por ejemplo, no necesitan software cerrado como MathLab para ilustrar lo aprendido en matemáticas, en especial con el nivel inicial que tienen los alumnos. En muchos casos no necesitan ni siquiera instalar programas específicos. Basta con que lancen Python —incluido en las netbooks—, instalen Ruby o R para comenzar a probar y a graficar funciones. De allí a la investigación científica están tecnológicamente a un paso<sup>79</sup>.

Siguiendo con el experimento, si se quiere lograr que la computadora mejore su capacidad de procesamiento, entonces se debería achicar el espacio que consume el sistema operativo y los programas residentes, para dar espacio en memoria RAM a las aplicaciones. En tal sentido, con Windows no se puede hacer mucho más que limpiar sus temporales y programas residentes. Windows, incluso en el mejor de los casos, posee una entropía gradual que obliga a acciones como defragmentar el disco, limpiar el registro, o quitar unas pocas actualizaciones que históricamente el sistema insiste en conservar. En el mejor de los casos, crece continuamente en disco y tarde o temprano debe ser llevado a ser formateado por los colapsados encargados de laboratorio, o enviado a Buenos Aires para que retorne a veces al cabo de dos años. La alternativa de reiniciar en *Linux* mostrará

79 Entre los muchos ejemplos que pueden recomendarse, observar los ejemplos del proyecto Jupyter, plataforma en Python que permite realizar simulaciones con algoritmos y matemáticas: <https://nbviewer.jupyter.org>

Ref: Anónimo (2016). *Jupyter - Open source, interactive data science and scientific computing across over 40 programming languages*. Recuperado de <http://jupyter.org>, [consulta el 25 de agosto de 2016]

una leve mejoría de velocidad, y pondrá al menos a correr a la computadora a una velocidad más cercana a la de sus verdaderos límites, es decir, con la memoria libre de programas residentes que no se sabe *efectivamente* qué están haciendo allí. Velocidad que quedará igual: Linux *no se pone lento con el tiempo*.

El otro experimento tiene que ver con el peligro que representan los programas que se obtienen en la Web. Como pregunta de investigación se puede apuntar a la incógnita respecto de cómo llegaron a infectar a la *netbook*. El trabajo puede enfocarse en analizar los procesos que realiza el usuario, y llamar la atención sobre la diferencia entre bajar indiscriminadamente de Internet, y la conveniencia de usar los repositorios automatizados de Linux, donde todas esas piezas de *software* están verificadas, centralizadas y auditadas. Los alumnos tienen internalizados parte de estos conceptos por estar presentes en sus celulares, dispositivos con Android y IOS, inspirados ambos sistemas operativos por el mismo proyecto madre de Huayra, Debian.

Si al cabo de estos pasos el alumno consigue *scrollear* (subir y bajar) con más soltura por la línea de tiempo de Facebook, bien por él: se lo ha ganado. Sin embargo es de esperarse que el alumno entienda que es perfectamente capaz tanto de construirse a medida su propio equipo, como de estirar sus límites tanto como sea posible.

## B.5. La CLI como protolenguaje de programación

Es interesante el hecho que recientemente Inglaterra haya incorporado la programación en la espacio curricular de Informática y elevado la misma a la categoría de ciencia<sup>80</sup>. El periodista Ian Livingstone (2012) lo explica de la siguiente manera:

"Vivimos en una era en la que la informática se ha convertido en el nuevo latín. Así como el latín ha sustentado tantas cosas, la informática no trata únicamente acerca de la programación: hablamos de pensamiento computacional, de resolución de problemas, análisis, física y creación de código. De la creación de contenido digital y de la propiedad intelectual. De la creación de valor en la economía digital."

De hecho en Inglaterra (no en el Reino Unido) se habla desde hace tiempo de quitar *Office* de los currícula. Es decir, darlo por sabido o ponerlo como materia coprogramática. Una postura que suena muy brusca para las aulas nuestras, es

80

- Livingstone, I. (2011). *Computer science is an essential part of our children's education*. Recuperado de <http://www.independent.co.uk/opinion/commentators/ian-livingstone-computer-science-is-an-essential-part-of-our-childrens-education-6268915.html>, [consulta el 15 de junio de 2012].
- Livingstone, I. (2012). *Teach children how to write computer programs*. Recuperado de <http://www.theguardian.com/commentisfree/2012/jan/11/teach-children-computer-programmes1>, [consulta: 6 de Octubre de 2013].
- Se sugiere la revisión del análisis mencionado, efectuado por el periodista Enrique Dans, y cuya traducción puede recuperarse de la siguiente ficha:  
Dans, E (2013). *La informática se incorpora como ciencia al bachillerato inglés*. Recuperado de <http://www.enriquedans.com/2013/01/la-informatica-se-incorpora-como-ciencia-al-bachillerato-ingles.html>, [fecha de consulta: 6 de octubre de 2015].

sin embargo lo que se hacía hace aproximadamente 20 años, en el salto que hubo de las micro computadoras a las PC, el equivalente tecnológico de la explosión de la vida en el Cámbrico.

Permítame una breve introducción al tema, hacia el año 1988 se enseñaba *Basic* en las pocas escuelas que podían permitirse el lujo de contar con Microcomputadoras *Talent*, *Commodore* o *Spectrum*. En realidad, se enseñaba programación, simplemente porque no había un concepto establecido de que pudiera "existir" las herramientas Procesador de Textos. Se enseñaba a programar pequeñas herramientas, porque las herramientas no existían. En la memoria colectiva de los informáticos, Estas computadoras se asocian usualmente con los viejos juegos de 8 bits, y naturalmente, *Basic* y algo de C, ambos lenguajes actualmente superados por opciones currícula convenientes en el aula, como las mencionadas en los capítulos anteriores.

La homologación en las parrillas nacionales y el auge de las PC trajo de algún modo la presencia del procesador de textos reemplazando a las máquinas de escribir, y a la hoja de cálculo. Programación quedó a criterio de cada escuela, generalmente en forma de taller y actualmente unas pocas escuelas compiten en las Olimpíadas Nacionales de Informática.

El consejo ingenuo aquí sería incluir de alguna manera elementos generales de algoritmos y de programación en el aula. Sin embargo, el problema inmediato que sobreviene a continuación se refiere a los recursos humanos: es muy difícil lograr que los docentes de informática enseñen programación en el aula o comiencen proyectos relacionados. Además del componente motivacional trabajado en capítulos anteriores, se pudo constatar serios inconvenientes para aplicar en la escuela los conocimientos aprendidos en el nivel Superior y Universitario, especialmente programación. Ello, a pesar de haber aprobado en algunos casos cinco niveles de programación y materias relacionadas como bases

de datos o sistemas de información. Por otra parte, también es muy difícil encontrar programadores que deseen dar clases. Hay una necesidad muy grande de programadores en el país: la mayoría se encuentra trabajando. Los que no se encuentran empleados, tienen dificultades para bajar al nivel de los alumnos, como fue ya mencionado en los tres tipos de exigencia de dualidad durante el capítulo de Análisis de Datos.

Ya habíamos tratado el problema del condicionamiento de las interfaces visuales en el apéndice A.4.4. de manera que como alternativa a la propuesta anterior, se sugiere incluir algunos elementos de razonamiento textual, sin *interfaces* gráficas, tales que permitan al alumno adoptar el razonamiento científico en tramos simples, inferencias rápidas de un medio abstracto con pocos pero sólidos elementos al alcance, con capacidad de prueba → efecto con resultados inmediatos: una forma de protoprogramación.

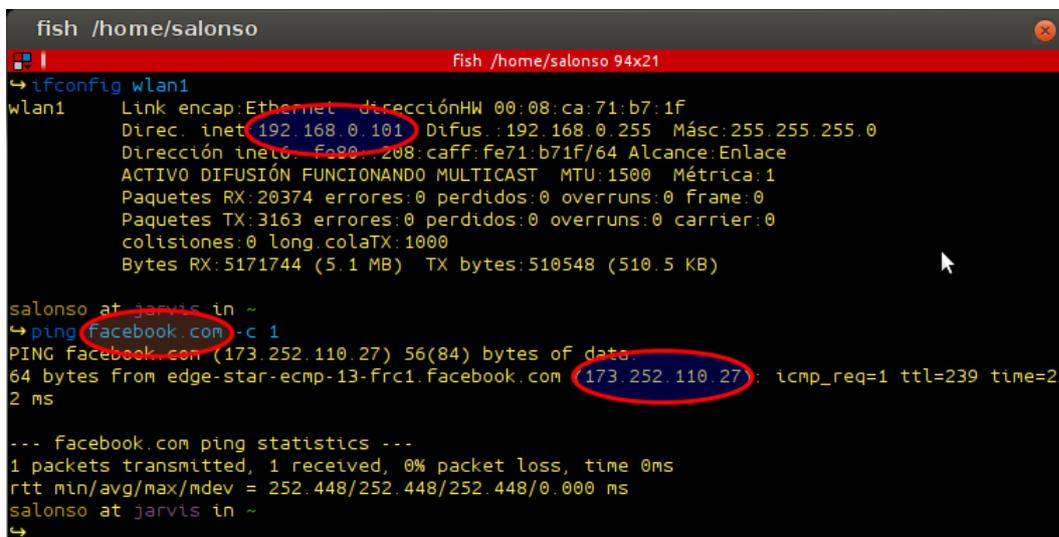
¿Es posible lograr esta propuesta, sin llegar a incorporar programación en las aulas? ¿Dónde se encuentra semejante herramienta?

En realidad se encuentra debajo de nuestra propia nariz. Todas las computadoras poseen una *interfase* auxiliar que permite llegar a lugares donde la capa visual no llega. Se trata del MSDOS en Windows, o de la Terminal en Linux, ambas conceptualmente similares. A estas *interfaces* se las llama CLI<sup>81</sup> o *Command Line interfase* (*interfase* por Línea de Comandos). Se diferencian de la programación tradicional porque están compuestos por varios pequeños programas que ejecutan instrucciones y devuelven un estado, tal que puede ser leído por otro programa, o por el usuario. Estos programas son como pequeños *bots* o "robots por *software*". Por su tamaño, raramente fallan, y de alguna manera

81 *Command Line interfase*: Esta expresión refiere típicamente, el MSDOS de Windows, a la herramienta Powershell, o la Terminal de Linux / UNIX.

siguen algunos elementos de la filosofía *Unix*, que ya se ha mencionado y que se puede resumir a nuestros efectos de la siguiente manera:

- Programas que hacen una sola cosa y que lo hacen bien.
- Programas que trabajan en conjunto.
- Programas que reciben y devuelven cadenas de texto, porque estas son una *interfase* universal.



```
fish /home/salonso
fish /home/salonso 94x21
↳ ifconfig wlan1
wlan1  Link encap:Ethernet direcciónHW 00:08:ca:71:b7:1f
      Direc. inet:192.168.0.101 Difus.:192.168.0.255 Másc:255.255.255.0
      Dirección inet6: Fe80::208:caff:Fe71:b71f/64 Alcance:Enlace
      ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
      Paquetes RX:20374 errores:0 perdidos:0 overruns:0 frame:0
      Paquetes TX:3163 errores:0 perdidos:0 overruns:0 carrier:0
      colisiones:0 long.colaTX:1000
      Bytes RX:5171744 (5.1 MB) TX bytes:510548 (510.5 KB)

salonso at jarvis in ~
↳ ping Facebook.com -c 1
PING Facebook.com (173.252.110.27) 56(84) bytes of data:
64 bytes from edge-star-ecmp-13-frc1.facebook.com (173.252.110.27): icmp_req=1 ttl=239 time=252 ms

--- Facebook.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 252.448/252.448/252.448/0.000 ms
salonso at jarvis in ~
↳
```

Ilustración N: Ejercicio identificando la dirección del Load Balancer de Facebook

Valga el siguiente ejemplo: un ejercicio que realizo durante la primer clase, consiste en sabotear de varias maneras Internet antes de que los estudiantes ingresen al aula. La actividad consiste en apagar el *router*, desenchufar cables al azar del concentrador ("*switch*") y, como un *gremlin*, desconfigurar casi todo. Una vez que los estudiantes ingresan al aula, y descubren rápidamente que no hay Internet, se les propone "arreglarlo".

Se les indica que abran el programa MSDOS (o la Terminal Linux), y que dentro de esa "aterrorizante" pantalla negra, identifiquen su dirección de IP (ifconfig, ipconfig). Luego, mediante una pequeña utilidad llamada *ping*, se los conduce a descubrir si su propia computadora es la única sin servicio. A partir de allí, detectar hasta donde llega la conexión. Paulatinamente y "lanzando pings", van descubriendo en qué parte de la telaraña están atrapados, y así van restableciendo los puentes hasta algún objetivo en concreto, por ejemplo en Facebook.

Durante esta actividad se promueve el trabajo en red sinérgica: los alumnos necesitan saber el estado de las otras computadoras: los alumnos se consultan entre sí, ya que cada *ip* en cada máquina es distinta, y en cada ella se debe emitir el comando *ping*. Las desconexiones, al igual que en una empresa o que en un barrio, no afecta a todos por igual. Una vez que los estudiantes logran verse entre ellos dentro de la red local, comparten sus estados para detectar "hasta dónde pueden salir", a fin de analizar la posibilidad de traspaso hacia el exterior.

Cabe suponer un bajo entusiasmo luego de semejante primer clase, al estilo de una "terapia de choque". Sin embargo, finalizada la hora, hay una manifiesta alegría por haber podido reparar la conexión, con una escasa intervención docente.

El ejercicio apunta a lo siguiente:

- concientizar que las *interfases* vistosas y modernas no ayudan a descubrir un problema vigente en el sistema y, menos aún, a resolverlo;
- la trama estructural de los dispositivos tecnológicos es efectiva pero abstrusa;
- para algunas personas, reparar cosas es divertido;

- un poco de razonamiento y sentido común ayudan a implosionar las lagunas cognitivas; y
- la tecnología en cuanto trama estructural libre de sus *interfases* patentiza la necesidad del estudio y la indagación constante de sus posibilidades.

Finalizado el cuatrimestre, aquellos alumnos que se acercan a preguntar sobre las particularidades de la carrera de sistemas, lo hacen auténticamente entusiasmados, con una noción bastante clara de lo que pueden esperar a futuro.

## **B.6. Hardware Libre**

El artículo antes referido de Livingstone (2012) efectúa el siguiente comentario:

"Enseñamos Física a nuestros alumnos no porque estemos esperando que se conviertan en físicos, sino porque viven en un mundo determinado por las leyes de la Física. Exactamente la misma razón por la que deben aprender Informática: porque viven en un mundo rodeados de máquinas programables".

La programación de máquinas no es tan difícil como era antaño<sup>82</sup>. En las mismas escuelas del Reino Unido los alumnos utilizan pequeñas plaquetas compuestas por un procesador, memoria, varias *interfases* de salida a teclado, monitor, puerto serie, *usb*, etc, llamadas *Raspberry*, vendidas por la *Raspberry Pi Foundation*, entidad mantenida por la Universidad de Cambridge, a un costo de U\$S 35. Las especificaciones son libres, de modo que pueden fabricarse también en el país.

82 Reflexión: la expresión "antaño" en informática bien puede referirse a apenas a cinco años atrás.

Idealmente pensadas para autodidactas, estas pequeñas plaquetas cumplen el sueño de todo profesor de tecnología que se encuentre interesado en motivar a los alumnos creando autómatas simples, detectores, sustitutos simples de celular, controlador domótico, y otras aplicaciones. Con el tiempo se ha creado un gran ecosistema de entusiastas alrededor de este *hardware libre*.

## **Apéndice C: El pensamiento estratégico necesario en las carreras tecnológicas**

### **C.1. Introducción**

A lo largo del capítulo 3, y específicamente en el sub capítulo 2.3.5. "Impacto sobre los alumnos" se trató el problema del docente que encerrado en un bucle de des-aprendizaje dentro de la escuela, coarta sus posibilidades de elección en cuanto a otros lugares de trabajo. Se llegó a un adecuado nivel de comprobación de este tema en varias entrevistados en donde los entrevistados manifestaban problemas para lograr la inserción fuera del ámbito escolar. Finalmente, durante la conclusión se propusieron diversas medidas paliativas para reubicar y acompañar al trabajador docente en una carrera profesional caracterizada por diversos desafíos en cuanto al enfrentamiento al cambio y a la complejidad.

Sin embargo, estas medidas se han de tomar cuando el problema ya se encuentra instalado.

En un ciclo ideal, el egresado de sistemas debería haber hecho su aprendizaje en las empresas, para después aplicar lo aprendido en las escuelas. Como se puede observar en gran parte de la población considerada para su estudio, las historias de vida se han dado al revés, lo cual no es problema en varios casos donde se demostró que el grado de contento y de retorno recibido es satisfactorio, medido por niveles de CM (Componente Motivacional) equiparado a niveles similares de VD (Vocación Docente) y de VE (Ventajas de trabajar en Escuelas). El presente apéndice que nos ocupa, refiere a la parte de la población estudiada que posee bajos niveles de estas tres variables, y no considera a los que se encuentran atrapados por RC2, es decir resistencia crónica más a la complejidad que al cambio, tema introducido en los capítulos 2.1, 2.3, 3, 3.1.1 y

3.1.6, sino a aquellos donde se detecta RC1, factores actitudinales para simplemente oponerse a él, y que redundan en el mismo problema: enquistamiento dentro del circuito educativo.

Acompañando esta demarcación, se vuelve al problema planteado en la primer hipótesis, respecto de la diferencia de la realidad que encuentra el egresado. Es decir, cuando aspirante calcula que una vez egresado, seguirá usando las mismas aplicaciones aprendidas en el nivel Medio, impartidas por sus profesores (Windows y Office). El siguiente nivel educativo supone unas cuantas incomodidades pasajeras, tras lo cual se recibe, y donde, a menos que tenga un plan para conseguir y mantener un empleo, termina entrando en el sistema educativo.

Hecha esta introducción, se pone a consideración el siguiente plan estratégico a aquellos que inician, que están intentando reinsertarse, o que no han pasado por el plan de profesionalización de carrera propuesto para directores en la Conclusión del trabajo.

Los seres humanos somos seres extraordinarios en comparación con los demás animales en el sentido que, dentro de ciertos límites, somos capaces de adaptarnos y reprogramar en gran medida nuestro árbol de conceptos. Ello sucede muy rápidamente dentro de una misma generación, sin necesidad de esperar grandes cambios evolutivos. No obstante, readaptarnos a un orden imperante requiere de una estrategia. A tal efecto es menester balancear las urgencias diarias respecto del rumbo trazado.

Senge indica:

"Desde mi comienzo en el MIT, me atrajo el trabajo de Jay Forrester, un pionero de la informática que había cambiado de especialidad para desarrollar lo que él denominaba *dinámica de sistemas*. Jay sostenía que las causas de muchos

urgentes problemas públicos, desde la decadencia urbana hasta la amenaza ecológica global, residían en las mismas políticas bien intencionadas que procuraban resolverlos. Estos problemas eran en realidad "sistemas" que tentaban a los funcionarios a actuar sobre los síntomas, no sobre las causas subyacentes, lo cual producía beneficios a corto plazo y perjuicios a largo plazo, y alentaba la necesidad de nuevas intervenciones sobre los síntomas" (1992, pp. 25).

Por esta razón el estudiante no puede dejar que la urgencia de lo cotidiano lo termine ahogando la visión ideal que tiene de sí mismo. En algunos casos el profesional tiene el capital o la ayuda necesaria para lograrlo, pero en otros casos, lo único que tiene es su capacidad de auto organización. Por lo tanto, la manera de mantener el rumbo es *establecer estrategias previas*, ordenarse a sí mismo antes de atacar un problema. Lograr la libertad de hacer lo que uno quiera, es un tema de ordenarse uno mismo, sin por ello dejar de lado las urgencias cotidianas. O como dice Morin (1994, pp. 151), "en la muy alta complejidad, el orden se vuelve libertad, y el orden es mucho más regulación que restricción".

El caso se puede ilustrar cuando el curriculum de vida adolece de experiencia, y no se poseen contactos. Sin embargo todo esto no es tan menesteroso; antes bien se trata de capitalizar estos elementos y reenfocarlos estratégicamente a través de una reingeniería de recursos, todo esto intrínsecamente conectada a su red social. Lo que se traduce en:

- a) preparar cuidadosamente lo que la gente va a ver cuando entre al perfil;
- b) nivelar hacia arriba, efectuando contactos con gente relevante;
- c) borrar toda persona cuyos comentarios son banales o hacen perder el tiempo;

- d) participar en grupos donde otros profesionales discuten temas atractivos y que involucren el perfil del estudiante fuertemente asociados a la carrera que sigue; y
- e) preguntar mucho, pero nunca si la respuesta está al alcance de *Google*; por mencionar algunos *tips* altamente recomendables.
- f) ayudar a otras personas

A menudo el estudiante plantea: "¿cómo hago para encontrar trabajo?". Bien, pues en tecnología, uno no va a buscar trabajo a las empresas, sino que las empresas lo buscan a uno. Los *headhunters* (los cazadores de talentos) primero miran el curriculum, luego chequean las intervenciones en torno a temas profesionales, en redes sociales profesionales como *LinkedIn*.

Tampoco los informáticos entienden que así como cambia el modelo de venta (basado en servicio en lugar de basado en el producto), también es distinta la manera en que las empresas captan este tipo de perfiles. Por ejemplo, cuando se trata de buscar programadores, las empresas grandes que usan *software* libre suelen mirar en los "créditos" (archivo *Credits*) de los programas liberados, y revisan listas de personas que colaboran con código fuente, simplemente accediendo al repositorio público. Seguidamente suelen chequear la madurez que manifiesta en las redes sociales informales (*Facebook*, *Twitter* y otras). En el caso de compañías de software cerrado, se presta especial atención a grupos de usuarios, tales como *Microsoft User Group*, *Oracle User Group*, etc.

Sin ánimos de brindar recetas pero sí de sugerir vías recomendables, se sugiere:

- a) cuidar qué tipo de *curriculum online* se socializa (además deben estar debidamente actualizados);

- b) si se carece de experiencia demostrable, capitalizar los "tiempos muertos" estudiando lo que requiere el mercado, y capacitándose luego mediante tutoriales y experimentos en casa;
- c) ser absolutamente cuidadoso y coherente con aquello que se escribe, publica o comparte;
- d) entender que el *software* se rodea de un ambiente especialmente meritocrático; y
- e) el prestigio lo aporta la calidad de las opiniones y de código que sus practicantes liberan a la comunidad.

De esta manera, los estudiantes (egresados o no) pueden acometer la resolución de la pregunta fundamental, a saber: "¿cómo hago para que me descubran?", comenzando por entender que las redes sociales sólo sirven si se hace de ellas un espacio de automercadeo, que puede medirse, por un lado, gracias a la actitud de servicio que se tiene frente a los demás, como mediante el intercambio honesto de conocimiento e impresiones con los pares.

La experiencia de campo también puede obtenerse previamente y no hace falta haber sido empleado. Muchas veces el novato puede mejorar sus competencias arreglando productos o brindando servicios en sistemas de escasa complejidad que resuelvan *problemas reales de personas reales*, adquiriendo así la proyección que necesita; por ejemplo, mantener una página *web* para alguna entidad sin fines de lucro o programar sistemas simples que ayuden a sus compañeros de trabajo. Idealmente basta que el interesado se baje el código fuente de algún programa libre que use normalmente, y que colabore, aunque sea con las traducciones. Estas acciones le aportan puntos de mucho prestigio en el curriculum. La única condición de estas acciones es que en lo posible, se

despeguen del clásico servicio técnico y venta de hardware, áreas por demás competitivas, y que no requieren del aprendizaje de una carrera.

Siguiendo esta idea, otro aspecto es considerar seriamente el trabajo de voluntariado. Si bien este tema en Argentina no posee una especial significancia, en Europa se toma muy en serio, al punto que las empresas inclinan mucho la balanza durante la elección del personal. De acuerdo a este criterio, a veces es preferible tomar una buena persona antes que otra extremadamente competente. Pertenecer a una ONG inclina en mucho la balanza al momento de cambiar de empleo.

Para los iniciados en busca de crearse curriculumms honorables, hay varios repositorios públicos de proyectos donde se puede colaborar con código fuente, traducciones, o simplemente reportando errores. Lo ideal es buscar proyectos activos y actuales, revisando las fechas de *releases* y de cantidad de *commits* al mes, lo que da un estado de salud de la comunidad que lo rodea<sup>83</sup>.

Finalmente, se ha llamado la atención durante el análisis de datos respecto de la importancia de la automatización de tareas recurrentes en las escuelas. Esta forma de trabajar es propia de otros ambientes a considerar seriamente por aquellos docentes que prefieren no tratar directamente con el usuario final. Tema difícil en las escuelas, donde precisamente, todos son usuarios finales, y en donde la actitud de servicio es especialmente importante.

De acuerdo a las entrevistas realizadas, y no se incluye a los profesores por la razón que ellos se sienten cómodos en la escuela, es frecuente la queja en los sujetos entrevistados, docentes que imparten clases o que están a cargo de

83 Anónimo (2016). *Project showcases - Browse interesting repositories, solving all types of interesting problems*. Recuperado de <https://github.com/explore>, [consulta el 24 de junio de 2016].

laboratorios de informática, respecto de la exasperación que producen el exceso de consultas triviales. Bien, para ellos, la practica constante de la automatización de tareas rutinarias es la puerta para emplearse en instituciones y empresas que cuentan con "ecosistemas" relacionados con servidores, y con especialistas en lugar de personas; ambientes tales como Infraestructura, Tecnología, Comunicaciones o Desarrollo, más adecuados para estos perfiles que se caracterizan por lograr mejores producciones cuando los dejan estar más concentrados en problemas complejos.

Las herramientas que se requiere aprender en este punto son las CLI (Command Line interfase) mencionadas en el Apéndice B.5, algo de base de datos, alguna forma para permear datos con el exterior mediante formularios, informes: PHP, HTML, algo de Ruby o Python en el caso de herramientas libres, Sharepoint o ASPX en el ambiente de Microsoft, y alguna máquina que pueda hacer de base para los primeros despliegues. De esta forma, se prepara el camino para salir de la escuela y profesionalizarse en el ambiente para el cual el docente informático, en definitiva, se preparó durante varios años.

## **C.2. Filtros en flujos de comunicación**

Hace ya muchos años en términos computacionales —es decir, 25 años atrás—, se introdujo en los *motherboards* de las computadoras, un pequeño chip denominado DMA (*Direct Access Memory*). Los ingenieros, diseñadores y fabricantes decidieron instalar un pequeño dispositivo en el *bus* de datos (pequeñas "autopistas" que se ven dibujadas en los circuitos integrados de silicio), tal que concedieran *automáticamente* permiso a la escritura en memoria o en disco a datos de sonido y vídeo, bastante irrelevantes para los procesos lógicos y

generadores de incontables interrupciones (IRQ, *Interrupt Requests*) del procesador. Este DMA permitía que la CPU no tuviera que supervisar pormenorizadamente cada dato y que pudiera concentrar sus procesamientos en aquello que resultara de mayor relevancia operativa. No obstante, estos datos quedan marcados como de *baja prioridad*, es decir que en cualquier momento pueden ser interrumpidos para el paso de instrucciones urgentes, según los requerimientos del usuario (la prioridad más alta en todo sistema).

Los seres humanos también tenemos una suerte de DMA. Por ejemplo los llantos de un bebé disparan varias urgencias atávicas que sólo hallan su término cuando el llanto cesa. Es decir, por términos de supervivencia el DMA humano no es perfecto, sino una brújula que se polariza al menor estímulo. En los tiempos modernos en los que vivimos, la asignación de prioridad se confunde a menudo, y tiende a elevar las motivaciones *secundarias* a la categoría de necesidades *primarias* (concepto básico de *marketing*). La comunicación es así, es prioritaria. El lector puede comprobar la veracidad de lo expuesto midiendo el tiempo en segundos (al menos 40 segundos) que se demora en volver a concentrar cada vez que llega un mensaje SMS, e-mail, llamada telefónica o de otra clase. Llevado este tema al plano consciente, de manejar una mínima inteligencia emocional y una disciplina equivalente, se puede programar el cerebro para filtrar lo útil de lo valioso, lo urgente de lo importante (nuevamente, los cuadrantes de Covey).

Paralelamente, en el capítulo anterior habíamos mencionado que es menester para el docente el planteo de una estrategia de tal modo que le permita liberarse de la escuela si le place. La focalización en el objetivo tiene que ver primero con poner cuidado en no subestimar lo complejo, lograr la realización personal mediante el título o "ser alguien en la vida" —algo que se pronuncia y que se recita con frecuencia, pero que sin embargo es muy complicado de lograr. Nuestro DMA se confunde con el elemento distractor por excelencia que es la

comunicación, la cual en exceso satura la capacidad de elección. Por lo tanto, el distractor sería cualquier cosa que interrumpa el flujo atencional del sujeto o que le haga perder el foco atencional. En tal sentido, la abstracción es una especie de identificación de factores operacionales interconectados que dan sentido a una estructura que se presenta como sistema y, a la vez, como un todo complejo.

Dicha situación se patentiza claramente en la instalación y el mantenimiento permanente de dos filtros (una suerte de disgregación del DMA mencionado). El primer filtro separa la información de los datos inútiles. Estrictamente hablando, se trata de la capacidad de determinar cuándo una información "parece" que es información, pero que debe ser degradada a la categoría de mero dato. Las páginas web de Internet se convierten en información relevante cuando son convalidadas en función de criterios consensuados, y con indicadores precisos de contrastación (coherencia con lo encontrado en la base empírica, consistencia lógica con los principios y conceptos sostenidos, convalidación de la comunidad científica internacional, entre otros). El segundo filtro refiere al inmenso ruido digital que hacen los demás seres humanos, *mientras están haciendo otra cosa* en Internet. Ello se muestra a las claras a través de comunicaciones distractoras.

Por alguna razón, las opiniones de las personas son un tipo de *metadato* (reflexión o análisis acerca de una información asociada a un dato) que adquiere condición de información en forma inmediata, generalmente sin contrastación ni proceso de validación alguno. Sin embargo, los conceptos que se vierten en las comunicaciones sólo sirven como idea para empezar a buscar. De otra manera son sólo *streamings* (flujos) y *broadcastings* (propagaciones) de banalidades, que tornan aún más oscura la complejidad y hacen que esta se vea aún más compleja.

Estos distractores deben ser identificados en forma *consciente* en el DMA humano para que sean redirigido al espacio *inconsciente*. El aspirante al mundo de los sistemas debe ser especialmente hábil para autoextraerse de los sistemas

que facilitan en grado sumo la comunicación, tales como grupos de Whatsapp, Telegram y equivalentes, si quiere lograr un proceso reflexivo. Tal como se ha mencionado en el Capítulo 3.1.6 "El bloqueo", el lector debería considerar la revisión que hace Gardner respecto de la inteligencia intrapersonal. O como menciona un *meme* de procedencia desconocida en Internet: "He encontrado la forma de hacer dinero con Facebook", tras lo cual agrega: "he cerrado mi cuenta".

## **C.2. Tratamiento terapéutico en docentes de tecnología para afrontar cambio y complejidad**

La decisión de abordar el conjunto de preguntas originales de investigación del presente trabajo tienen su origen en la conclusión que quedó tras un plan de mejora de servicio realizado en la Escuela Normal 9-002 "Tomás Godoy Cruz" desde el año 2009 al año 2014, fecha en los equipos fueron finalmente dados de baja o repartidos en función de la llegada de las *netbooks* del proyecto Conectar Igualdad.

Esta mejora consistió en una migración de los equipos de laboratorio de informática al sistema operativo Linux. Durante este plan de mejora, se planificó cuidadosamente cada etapa para que el cambio fuera gradual y paulatino para todos los usuarios, por lo que hubo tiempo para observar, corregir y accionar en consecuencia. Las características de este bucle compuesto por sucesivas acciones correctivas tuvo varios elementos en común con la visión técnico-científica propuesta por Kurt Lewin para su concepción del diseño investigación-acción, rescatada por Hernández Sampieri, Fernández Collado y Baptista Lucio (2010, pp. 510) y planteada en forma de niveles a lo largo de toda la obra de Paul Watzlawick (1992).

La migración mencionada surgió como propuesta de salto transversal para resolver varios problemas de nivel 1 aparentemente irresolubles. Esto es, constantes fallas en el servicio producidas por exceso de carga en el laboratorio de informática: en esencia, 1430 alumnos y profesores que compartían 13 computadoras. Se debe mencionar que el laboratorio se encontraba enmarcado bajo un proyecto propio denominado "Ciber Educativo", que consistía en asistencia de 8 a 18 hs a planta docente, profesores y alumnos, en ese orden de prioridad, con las observaciones que el lector juzgue adecuadas. Entre las órdenes

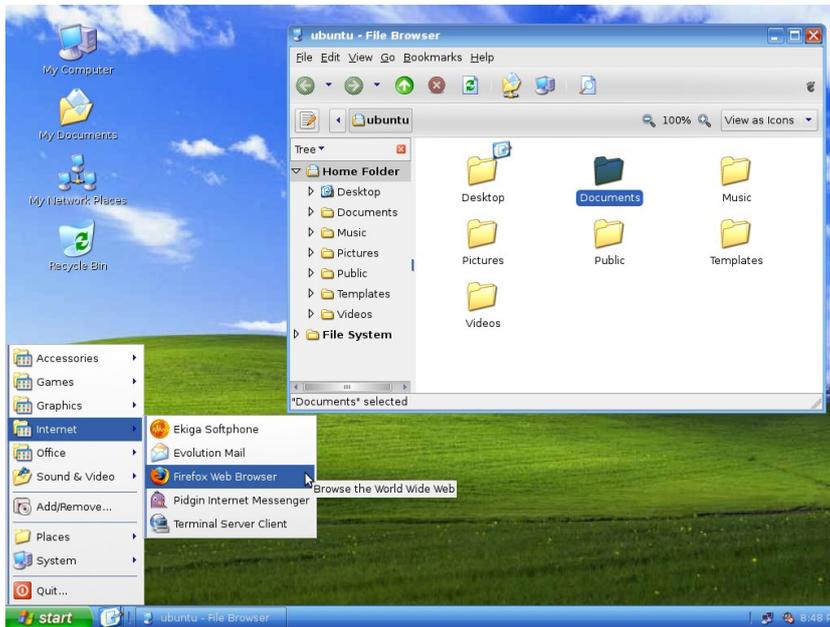
recibidas, acordes a la cultura institucional, se incluía hacerles el trabajo a los administrativos que no podían o no querían usar computadoras. Los actores mencionados eran asistidos en la confección de sus trabajos, y la cantidad de computadoras es determinante para prestar un buen servicio. Estos equipos hasta el momento en que fueron dadas de baja en el mencionado 2014, funcionaban a toda hora.

Debido a la cantidad de macro virus de Word bajados por e-mail y cargados por pendrives, sumada a la inestabilidad de Windows XP, todo el tiempo dos o tres máquinas estaban en reparación, de manera que las colas de espera para usar las máquinas se hacían interminables, y la presión sobre los cuatro encargados de laboratorio era permanente. A pesar de contar con una fuerte batería de antivirus y de mecanismos de *freezados*, volcado de imágenes por red y *backups* automatizados, tarde o temprano éstos paliativos eran sobrepasados, o en el mejor de los casos dejaban tan lentos los equipos que las quejas eran permanentes. Los equipos no eran modernos: Pentium 4 con 512 MB de RAM, y no cumplían con los requerimientos recomendados para Windows Vista ni Windows Seven.

El salto transversal mencionado obedeció al planteo teórico de mejorar el servicio, como se ha mencionado, migrando a Linux, para lograr al menos una completa inmunidad a los virus.

Por tal razón se trabajó un diseño investigación-acción. Es decir, no había tiempo para realizar una investigación longitudinal de un año para aplicarla al siguiente. Se debía investigar a medida que se mejoraban las condiciones del servicio, y se debía realizarlo junto a los beneficiarios, que se convertirían en sujetos de estudio a medida que se fueran sumando iteraciones. Estas iteraciones a la manera de las acciones planteadas por Lewin, eran en forma de asistencia permanente antes y durante la migración. Estas asistencias se

realizaban de forma inmediata, no solo por la urgencia de lograr resultados, sino para evitar sumar quejas por reacciones colectivas al cambio, a las quejas normales y cotidianas.



*Ilustración O: Interfaz Gnome 2 modificada durante la migración mencionada. Aceptada por estudiantes y docentes. Rechazada por docentes de informática.*

Se debe considerar que tras aplicar los cambios, y tras unas tres iteraciones que tomaron aproximadamente 7 meses, entre febrero y agosto de 2009, el servicio de laboratorio mejoró en gran medida: se redujeron las esperas por máquinas disponibles. Por primera vez en la historia del laboratorio de informática las 13 computadoras estaban funcionando al 100% sin caídas ni cuelgues. Tras resolver algunas dudas iniciales, los usuarios hacían su trabajo rutinariamente y se retiraban con sus impresos a tiempo. Nadie se quedaba sin atender. Los alumnos especialmente, acompañaban el cambio intuitivamente, realizando muy pocas preguntas, para continuar trabajando sin mayores comentarios.

No obstante, esta solución fue parcial según el esquema de cambio propuesto por Watzlawick, puesto que incluía bucles emergentes de bloqueo en algunos sujetos. Se reiteró nuevamente la investigación (realimentación) con el objeto de entender, por encima de apreciaciones simplistas, la naturaleza de estos bucles de incompreensión. Así se pudo detectar que los sujetos bloqueados ante la interfaz Linux (Gnome) eran

- docentes, que tenían también problemas con sus propios Windows; y
- docentes de tecnología, que poseían un fuerte condicionamiento por el modelo de *interfase* propuesto por Windows.

En el primer caso, tan sólo adecuando la *interfase* para que coincidiera con sus supuestos, se logró un apreciable cambio a nivel 3: correcto uso de la interfaz *Gnome 2* de *Linux* (ver Ilustración O). Incluso varios docentes llevaban sus máquinas personales para que les fuera instalada la misma *interfase*.

Sin embargo, para el segundo caso (los docentes de Tecnología), no hubo manera de superar sus condicionamientos a las *interfases*. El punto de palanca utilizado (O'Connor y McDermott, 1998, pp. 292) que funcionó con los otros actores, no funcionó en este caso: los docentes no asistían a las capacitaciones ni se dejaban ayudar en tiempo real.

La ingenua conclusión preliminar consistió en que simplemente los docentes de Tecnología "no querían usar Linux". Sin embargo varias alertas se encendieron en entre las mismas observaciones. Una de ellas, como se ha mencionado, consistió en señalar en que los docentes con resistencia a Linux tampoco eran competentes usando Windows. Otra observación se basaba en la observación que cinco de los profesores de tecnología, cuando usaban unos pocos equipos dispuestos en el laboratorio con Windows, y se bloqueaban, se negaban a ser

ayudados. Otra observación consistió en tampoco podían usar correctamente las aplicaciones portadas. Por ejemplo no usaban Openoffice para Windows. Incluso hubo un caso de excesiva resistencia: a pesar de las capacitaciones ordenadas desde Dirección a principio de año, un docente se negó de plano a entrar al laboratorio, y decidió seguir enseñando a usar el procesador de textos... en el pizarrón del aula, dibujando *mouse*, teclas, regla y botones. Lo mismo con el "episodio del navegador", una crisis de nervios documentada por el mismo director, en el que el docente "no podía enseñar a navegar con *Firefox*. Solo podía utilizar *Internet Explorer*".

Semejantes observaciones daban lugar a que había problemas más profundos a explorar. Tan solo la paradoja de educadores que no se dejan enseñar constituyen todo un problema de investigación en sí mismo, ya desarrollado en el Capítulo 3 "El Lobo Solitario".

Continuando con el orden cronológico, seis años después a las observaciones mencionadas en el año 2009, sumadas a las entrevistas realizadas en la actualidad, y a la carga teórica empleada, se puede determinar que la paradoja mencionada no lo es tanto, si se sigue la acepción moderna de la Teoría de Sistemas interpretada por O'Connor y McDermott, quienes entienden a los sistemas como matrices elásticas, sobre las cuales todo intento de retirar un nodo irá acompañado de vectores del otro lado que harán fuerza por volver el nodo a su sitio (resistencia al cambio). También se entiende que los docentes de Tecnología se encuentran más expuestos a la contaminación del sujeto por parte de su objeto. Al estar conectados al sistema por muchos más nodos que una persona común, su desconexión implica un desarraigo a varios niveles medulares. Debido al plexo de referencialidad que han construido en su quehacer —en sentido heideggeriano— son víctimas del mismo (sistema), pero del que el sujeto es parte. Ahora bien, si se efectúa una relectura desde la TGS, el ensimismamiento del

*Dasein* (el ser humano) y sus modos de ser y de proyectarse en el mundo, cuando actúa según el "se" (*man*<sup>84</sup>), donde el ser se pierde en el conjunto de lo preestablecido como dado en su precomprensión de las cosas, actúa bajo sistemas de realimentación condicionados matricialmente por trampas que nutren su propio ensimismamiento y cerrazón (ya explicado en el cap 2.4 "Mecánica en lugar de Sistemas"), al que realimentarán cuando ejerzan la docencia en carreras afines. El problema sigue siendo entonces: ¿cómo escapar a la matriz siendo la matriz misma la que se autoconstruye y legitima como mundo y horizonte vital?

De esta manera, el punto de palanca en este nivel 2 es distinto para el caso de los docentes de Tecnología. Sólo puede darse de manera personal: no se puede entrar a la "matrix" (jugando con la idea de matriz de O'Connor y McDermott) a proponerles una reestructuración de sus modelos. Siendo ellos personal de sistemas, sólo pueden salir de la misma realizando una introspección precisamente a la manera de sistemas, es decir, al detectar

- que tanto se han dejado llevar por la mecanización;
- que tanto se han limitado a una sola *suite* ofimática gestáltica para solucionar problemas de tecnología aparentemente holísticos (desde su perspectiva irreductible), y de educación para con sus alumnos;
- que aquello que parecía un tratamiento holístico, no lo es tanto cuando
  - se desaprovechan los emergentes de conjunto (por ejemplo las redes sociales); y

84 Este *man* es el mismo modo de ser instalado como modo naturalizado por Wintel. Cfr. Apéndice A.4

- se abstiene de alterar las partes, reducirlas o experimentar con ellas para descubrir nuevos efectos en el todo, cuyos ejemplos serían desaprovechar los muchos lenguajes de programación pensados para estudiantes, ignorar la API (*Application Programming interfase*) de redes sociales, desconocer los SDK (*Software Development Kit*) de los celulares, las pequeñas SBC (*Single Board Computer*), o bien salir de lo técnico, esquivar los muchos trabajos de apoyatura que pueden hacer para las otras materias.

Proponer por tanto a los docentes una revisión de la lógica que subyace a sus conocimientos de base puede ayudar en gran medida a que revisen sus modelos de pensamiento. La capacitación en este sentido no es fácil, ya que requiere de un adecuado tratamiento, resumen y encapsulación de la bibliografía existente, la cual se encuentra al final de la presente obra. Los docentes de Tecnología deben ni más ni menos, que estudiar aquellos contenidos que no les fueron suministrados durante el cursado de sus carreras de sistemas. La presentación más adecuada de esta hipotética capacitación debería tener un formato de tipo "Teoría de Sistemas como Pensamiento Sistémico". En este sentido han habido capacitaciones de este tipo en la provincia, pero inadecuadamente impartidas desde consultoras a empresas, con orientación a mejoramiento de líneas de montaje de productos (como el modelo fordiano que se aplica en las aulas), en lugar de orientar al servicio, que es lo que en definitiva hacen los docentes.

## **Apéndice D: ¿Aprender a resolver problemas, o resolver los problemas de las empresas?**

En Argentina la figura del "pasante" en las empresas hizo que las instituciones educativas se prestigiasen en la medida que sus pasantes fueran contratados ávidamente por las mejores empresas.

En la década de los 90 se puso la lupa sobre el desfajase entre los — aparentemente— rudimentarios conocimientos adquiridos en la escuela, con respecto a las necesidades de las empresas, los adalides del éxito en aquella década neoliberal. Si nos remitimos a esa época, las empresas exitosas se convirtieron en modelos o referentes que, en ese contexto, dieron lugar a un replanteo profundo acerca de la finalidad de la educación: ¿para qué forman las escuelas? ¿Forman ciudadanos genéricos, que se adaptan al sistema?, ¿forman estudiantes que puedan insertarse en la educación superior, es decir, potenciales profesionales?, ¿forman seres críticos, autónomos y creativos?, ¿forman seres competentes capaces de insertarse en el mundo competitivo contemporáneo con alta calidad técnica y versatilidad ante el cambio constante? A riesgo de caer en un monólogo, los planes de estudios de las escuelas secundarias de ese momento, y de ahora también, muestran que merced a la sola enseñanza de ofimática, formamos apenas usuarios de oficina.

Esto condujo no solo a la reforma educativa de los 90, sino también a repensar el papel de la educación, el individuo y el Estado; el modelo de sociedad al que aspiramos como argentinos y el lugar de la Ciencia y la Tecnología, la I+D, configurándose la Ciencia como nuevo objeto de estudio, en el marco de la sociedad del conocimiento. Elementos que se enarbolan como ventajas competitivas y estratégicas para un mundo recién globalizado

Este dilema encontró su eco en las carreras de sistemas. Por ejemplo, se revisó si los lenguajes de programación estaban "oxidados" para la realidad del mercado. De esta manera se impulsó el aprendizaje de *Visual Basic* (Microsoft, cerrado, fácil de usar) por sobre el viejo lenguaje C (libre, abierto, complejo), debido a que el primero permitía rápidamente crear y diseñar maquetas de un sistema, en relación al segundo, que requería de una mayor minuciosidad en los algoritmos y la programación de sus partes. Los mismos estudiantes de la generación de quien escribe estas líneas, fuimos ingenuos promotores de la adopción por el novedoso *Visual Basic*, al exigir a los directores de carrera la imperiosa necesidad de "estar al día", priorizando una sobre la otra. De acuerdo a mis observaciones personales realizadas más tarde en ejercicio de la profesión, indican lo equivocado de mis exigencias durante la carrera, por cuanto la focalización de uso en *Visual Basic* en la carrera de Sistemas es potencialmente lisiante en los egresados.

Para señalar el hecho, se puede corroborar que al menos en los planes de la universidad, existe un mejor balance en los algoritmos elegidos respecto de aquellos años '90. Se trabajan C, PHP y Java, todos lenguajes gratuitos y abiertos que pueden ser utilizados tanto por línea de comandos como por interfases visuales, que proveen adecuadamente los rudimentos de lo que pueden encontrarse tras egresar: florecientes ecosistemas que surgen por doquier, en donde los informáticos viven muy bien de originales propuestas como Python, Eiffel, R, Erlang, Ruby y muchas otras opciones, en relación a los pocos y hegemónicos lenguajes de los '90.

Sin embargo se debe prestar especial atención a la invasión de propuestas mágicas traídas por empresas en los claustros: estas acciones equivalen a permitir a que un laboratorio privado dictamine las soluciones que deben administrar los futuros doctores de la Facultad de Medicina.

## **Apéndice E: Hurto de *software* en el aula**

### **E.1. Principio económico de la escasez**

Churión (2011) enuncia este principio de la siguiente manera: "la cantidad de bienes económicos disponibles son limitados, mientras las necesidades materiales de la sociedad son crecientes e ilimitadas". Si todo bien económico es escaso, entonces no todos tendrán acceso y se generarán diferencias entre los que lo logran y los que no.

Este problema de la accesibilidad se suma a otro que se da en los márgenes entre lo accesible y lo no accesible, que es el del acceso fraudulento. Usar un *software* de cientos o miles de dólares sin haber pagado por él, tiene un atractivo morboso de lo robado, es decir, tener aquello que no es propio pero gozar de él por el beneficio y placer que reporta tenerlo, sin haber erogado el equivalente energético, medido en dinero. Paralelamente, con las producciones de este *software* se espera alcanzar los mismos horizontes que otras personas que pagan por el mismo, sin tener que pagar, gastar, invertir, dar de sí o trabajar para obtenerlo. Ya solo este comportamiento legitima el hecho desde un punto de vista de supervivencia. Es decir, se suma que el objeto permite alcanzar logros que de otro modo resultarían inasequibles, y es precisamente ello lo que otorga satisfacción libidinal con relación al objeto —en sentido psicoanalítico tanto freudiano como lacaniano. Se puede aplazar la moralidad del asunto, si luego se amortiza en forma suficiente como para pagarlo *a posteriori*, algo que normalmente no ocurrirá.

## **E.2. Apología del hurto por parte del docente**

Generalmente el estudiante no sabe que el *software* que usa en sus equipos otorgados por el gobierno viene pagado de manera transitiva por sus mismos padres, quienes son contribuyentes del sistema. Los docentes no explicitan la procedencia de los programas y de los sistemas en uso. El aspecto ético permanece ausente del aula, y el jurídico, aplazado hasta que los estudiantes tengan empresas a las que puedan exigirles el abono de licencias.

La simple mención pasajera bastaría para desnaturalizar un hecho que se presenta como instalado y obviado en el uso de las tecnologías. A nadie se le ocurriría tomar una plancha de un escaparate de un negocio y salir con ella para usarla en su casa por el simple hecho de que puede hacerlo. Este ejemplo es tipificado de robo simple. Sin embargo resulta muy poco frecuente el hecho de que alguien reconozca que copiar un programa no liberado es un hurto simple. El docente de tecnología debería al menos mencionar este hecho y ofrecer una alternativa al respecto, debido a que cuando el alumno use cualquier otra computadora, sólo podrá hacerlo con aquellos dispositivos que aprendió. De otra manera, el docente es cómplice, instigador indirecto, responsable y culpable del hecho por omisión.

De la misma forma, en muy pocos casos se explicita en clase la distinción entre productos comerciales, gratis y liberados. De hecho los docentes generalmente no distinguen entre algo libre (abierto) y algo gratis. De esta manera, el docente deja a criterio del alumno el uso de programas y sistemas comerciales. Lo deja a su libre albedrío. Sin embargo, el albedrío se encuentra fuertemente condicionado tanto por las necesidades del sujeto como por su desarrollo de una moral autónoma madura y responsable; una moral que a menudo no está desarrollada ni siquiera en los adultos. Se encuentra también condicionado por lo

que conoce, o lo que único que aprendió a usar. En otras palabras, el docente no deja al alumno que elija basado en su libre albedrío: lo deja que elija basado en su *condicionado* albedrío.

Un hurto es un hurto y, éticamente, no importa que el programador sea un millonario lleno de dinero. Ha solicitado un canon por su producto, y está en su derecho reclamarlo. Toda otra argumentación es una argucia falaz para autojustificar conductas fraudulentas e inmorales.

Ni el alumno ni el profesor son héroes de la resistencia cuando ignoran explícitamente este hecho. El profesor se encuentra especialmente comprometido desde el momento en que distribuye apuntes en formato propietario (doc, docx), ya que cuando el alumno deba abrirlo en otra computadora que no sea su *netbook* (que trae el programa ya comprado por el gobierno), deberá pagar por su *Office* o violentarlo mediante algún programa "crackeador". Cuando el docente deja a consideración esta acción al alumno y se desliga del problema, discrimina tanto a los "éticos" que adoptan formatos abiertos, como a aquellos que no pueden pagar para abrir el documento.

Por supuesto, los formatos cerrados pueden abrirse mediante *Openoffice* o *Libreoffice*. Pero en su juego de mercado, *Microsoft* posee una larga lista de omisiones —y juicios— respecto de las escasas descripciones en los diccionarios de datos de sus productos, a fin de que los documentos posean deformaciones e inexactitudes cuando sean abiertos por otros programas.

Sin embargo, y a pesar de las ya muchas bromas que circulan por Internet y televisión por lo flagrante y explícito del hecho<sup>85</sup>, es llamativo lo poco que se trata

85 Análisis de Futurama: Sexta Temporada, Capítulo 25. Ficha disponible en

- Anónimo (2015). *Overclockwise*. Recuperado de <http://theinfosphere.org/Overclockwise>, [fecha de consulta: 20 de marzo de 2014].

el tema en la cátedra de Ética Profesional en las carreras de Sistemas, Informática y afines. La única deducción que puede obtenerse de semejante omisión es que no se plantea formalmente la pregunta en clase: ¿por qué las personas hurtan *software*?, o mejor aún, ¿por qué nadie se hace esta pregunta?

A riesgo de contestarme en un juego retórico, una pregunta que nadie hace, es aquella cuya respuesta es aceptada socialmente como "evidente". Es decir, cuando la respuesta es demasiado obvia, es implícita, es axiomática; y como todo axioma, se acepta como elemento fuera de discusión. Sin embargo, la naturaleza de los axiomas busca precisamente reducir lo complejo a unidades autoevidentes. El resultado termina en reducciones más hechas, con sesgos de partes que son tan irreductibles como valiosas por la información técnica y ética que suponen para el estudiante. Un fenómeno que puede resumirse en la expresión "una verdad incómoda".

### **E.3. Cleptoparasitismo**

Si la respuesta a lo antes expresado es demasiado obvia (los docentes acompañan el hurto de *software*), entonces la pregunta no está bien planteada. En todo caso podemos expresarlo en forma de pregunta de investigación: ¿por qué los docentes, en forma consciente o inconsciente, promueven el robo de *software*?

A tal efecto, desarrollemos la siguiente aseveración, que puede resultar como sorprendente por su brevedad: *las personas hurtan software porque tienen manos*.

- Análisis de La Familia Simpson: Temporada 11, capítulo 235. Ficha disponible en Anónimo (1999), *Discuss Grift of the Magi*, Recuperado de <http://www.imdb.com/title/tt0701101/>, [fecha de consulta: 20 de marzo de 2014].

Esta afirmación puede inferirse a partir de la explicación antropológica que hace Ciarlotti (2011), quien señala la relación que existe entre el bipedismo y el cleptoparasitismo interespecífico:

"Esta liberación de los miembros superiores fue, en su inicio, una adaptación óptima al bioma de la sabana; al poder marchar bípedamente y con los brazos libres, los ancestros del hombre podían recoger más fácilmente su comida: raíces, frutos, hojas, insectos, huevos, reptiles pequeños, roedores y carroña; en efecto, muchos indicios hacen suponer que nuestros ancestros fueran en gran medida carroñeros, y dentro del carroñear, practicaran la modalidad llamada cleptoparasitismo, esto es, robaban las presas recién cazadas por especies altamente carnívoras; para tal práctica, nuestros ancestros debían haber actuado en bandas, organizadamente".

De esta manera, se puede teorizar que el hurto sin dolo, de algo aparentemente inmaterial e inocente (como el *software*) es un acto atávico, incluso inconsciente (como el de los cleptómanos) que carece de una condena social consensuada. Llevado al terreno de la conciencia, no molesta mucho allí, pues la legalidad de ciertas formas de hurto se encuentra justificada incluso por la ley. Para ésta, el hurto de comida y sin dolo hacia los demás es un delito excarcelable, comprensible por la urgencia que demanda el hambre; es preferible a vivir el estado que sucede a la necesidad insatisfecha, que es el robo (hurto con violencia). Por tal razón, para el inconsciente, no hay diferencia entre *hurtar comida* y *hurtar "por" comida*.

De esta manera, la pregunta original debe replantearse respecto de si es *justo hurtar herramientas*. Esta discusión es probablemente más antigua de lo que se esperarí a simple vista. Muchas civilizaciones coincidían en despenalizar el hurto de comida tanto como castigar el hurto de herramientas.

En todo caso, las novedades que ha introducido la Edad Contemporánea han sido que, por un lado, el cleptoparasitismo ha pasado a ser intraespecífico, es decir, ya no hurtamos carne a los leones, ni piedras a las nutrias con las cuales romper nuestros mariscos. En el Antropoceno los seres humanos se hurtan entre sí; y, por el otro, las bandas parasitarias de recolección —a las que hace mención Ciarlotti— se han organizado de dos maneras:

- a) se encuentran aquellas que ponen a disposición de sus miembros los despojos que encuentran en las ciberesferas (edifican sitios de descargas y crean programas *parásitos* para bajar material con derechos de uso<sup>86</sup>); y
- b) otras bandas (en el sentido antropológico de agrupaciones de individuos) adoctrinan a sus miembros respecto del correcto uso del material encontrado: son las escuelas y las universidades.

La diferencia entre ambas bandas es que las primeras son ilegales. Las segundas enseñan a usar lo rapiñado.

Ahora podemos traer a colación la hipótesis  $H_{1,3}$

Guía: en referencia a hipótesis  $H_{1,3}$ : La exposición inicial a interfases enriquecidas sería directamente proporcional a la resistencia al cambio de frontend, e inversamente proporcional al aprovechamiento del backend subyacente.

Aquí se encuentra la figura del alumno "simbionte" con el *software* comercial: aquel que ha contraído el programa por exposición a sus profesores parasitados<sup>87</sup>. ¿Es una ventaja para el alumno la competencia y displicencia en el uso de

86 Como curiosidad a esta analogía con el cleptoparasitismo, en el protocolo Bittorrent existe la modalidad "sanguijuela", que es la actividad de *bajar* material sin *subir* nada a cambio.

87 La simbiosis es ventajosa para las partes, en tanto que una no subyugue a la otra. De otra forma se trata de parasitismo.

*software* robado? ¿Es una nueva habilidad evolutiva, heredada, la de robar *software* más velozmente, más subrepticamente? ¿no debería el alumno de sistemas aprender a bajar *software* que ya viene abierto, de donde puede aprender, en lugar de rapiñar piezas terminadas que no puede reproducir?

Esta suerte de dependencia del *software* cerrado ha sido advertida en varias ocasiones: muchas instituciones educativas a nivel mundial exigen a los docentes entregar su material en formatos abiertos como *ods*, *odt*, *rtf* y otros.

De la misma manera, muchos estudiantes introyectan esta dependencia del *software*, este poco control sobre las TIC, prótesis que extienden diariamente las posibilidades de los humanos, se adivinan como una imagen poco digna, y generan ruido en diversos foros.

Estos ecos en la web, estériles en apariencia, son jugosos puesto que proponen una discusión ética, la cual en teoría no admite puntos grises: se es moral<sup>88</sup> o no se es. En la práctica, la cuestión nunca queda sanjada: en la esfera privada (personal) y privada (empresarial) se usa lo que uno quiera o pueda pagar, y la ley de conservación de energía es la que decide en todos los casos.

#### **E.4. El software al rescate de las utopías**

¿Es entonces ésta, una discusión estéril? Sí, cuando una de las partes insiste en que la otra "tome conciencia". Hall (1994, pp. 325) apoya involuntariamente esta idea cuando señala que

88 Principio 2.02 del Código de Ética y Práctica Profesional de Ingeniería del *Software*. Anónimo (2015). *Software Engineering Code of Ethics and Professional Practice*. Recuperado de <http://www.acm.org/about/se-code>, [fecha de consulta: 21 de marzo de 2014].

[...] "existe una creencia implícita, extendida tanto en EE.UU. como en Europa, que la toma de conciencia conduce al cambio. Resulta, sin embargo, que mis propios trabajos, así como el del grupo de psicólogos transaccionales [...] demuestran que la toma de conciencia no conduce al cambio más que en situaciones muy particulares"<sup>89</sup>.

Si se tiene en cuenta que toda ética tiene una naturaleza utópica y que debido a su carácter deontológico, no habla del *ser* sino del *deber ser que aún no adviene*, toda labor que se emprenda resultará idealista. Por lo tanto es una decisión personal abandonar el intento, siguiendo argumentos como los argüidos por los psicólogos transaccionales como Paul Watzlawick (1994, pp. 343) que en virtud de sus años pasados en la segunda guerra mundial, profundiza la posición precedente<sup>90</sup> al decir:

"Personalmente, sé por triste experiencia que cuando se persiguen finalidades utópicas se acaba en osarios y campos de concentración. A partir del momento en que se sacrifica lo posible a lo deseable, se interna uno en una vía inhumana".

Sin embargo, lo curioso del caso, es que incluso en países capitalistas con influencia mercantilista, y a pesar de lo pragmático que conlleva el ejercicio de la tecnología, muchos informáticos toman el camino *utópico*. Por ejemplo, la Licencia Pública General (GPL) y la licencia Berkeley Software Distribution (BSD), que proveen de marcos legales para "regalar" el software, fueron creadas paradójicamente en Estados Unidos. Estas licencias amparan toda la tecnología que mantiene funcionando una Internet libre y gratuita para todos.

Con estas acciones de ejemplo, y volviendo a la vida a Oscar Wilde, podemos descubrir que es el *software* quien viene a rescatarnos de la triste opinión de Watzlawick:

89 Entrevista a Edward T. Hall efectuada por Martha Davis en *La Nueva Comunicación*, pp. 325.

90 Entrevista a Paul Watzlawick por Carol Wilder, en *La Nueva Comunicación*, pp. 343

"No vale la pena siquiera dar un vistazo a un mapa del mundo en donde no se incluya a Utopía, por cuanto es el país hacia donde la humanidad terminando llegando. Y cuando la humanidad arriba allí, mira alrededor, y buscando un país mejor, iza las velas y zarpa. El progreso es la realización de Utopías"<sup>91 92</sup>

Volvamos al presente: fácticamente, utópicos y pragmáticos conviven en el mismo mundo, más unidos e interdependientes de lo que se cree. El *software* libre es imprescindible cuando hace falta manejar cantidades descomunales de datos, distribuir carga o eficientizar al máximo los recursos en ciertos ambientes donde los números —por su tamaño— expresados mediante guarismos. Rovers marcianos, el Colisionador de Hadrones, grandes bases de datos como Cassandra, estructuras como Glusterfs, muchos proyectos científicos y también comerciales como *Google* o *Facebook*, por poner unos pocos ejemplos, son imposibles de concebir sin el intercambio de código fuente<sup>93</sup>.

La paradoja que surge es que los informáticos *pragmáticos*, aun despreciando el idealismo de las patentes libres, conviven y se comunican en un mundo utópico-comercial, gracias al ávido consumo de *software* social creado por *idealistas*, quienes les ayudan a llevar a su individualismo a nuevas fronteras.

91 *La cita corresponde a nuestra propia traducción del texto "A map of the world that does not include Utopia is not worth even glancing at, for it leaves out the one country at which Humanity is always landing. And when Humanity lands there, it looks out, and, seeing a better country, sets sail. Progress is the realisation of Utopias".*

De esta cita, puede encontrarse el original como libro liberado en razón de los años pasados desde el fallecimiento de su autor, y escaneado el texto original en el Proyecto Gutenberg.

92 Wilde, O. (1900). *The Soul of Man under Socialism*. Recuperado de <http://www.gutenberg.org/ebooks/1017>, [fecha de consulta: 22 de agosto de 2016]

93 Se sugiere la revisión del proyecto de computación distribuida para apoyo a proyectos científicos, que mantiene la Universidad de Berkeley en el Proyecto Boinc.

Basta con observar a aquellos usuarios pendientes de su celular con *Android* (basado en *Linux*) o *IOs* (basado en *FreeBSD*), que confían toda clase de acciones a esta combinación de programas abiertos y cerrados<sup>94</sup>.

Promover y propiciar la apertura de los *softwares* es una realidad posible en tanto se beneficien todos y "especialmente" cada uno. Si sólo se benefician aquellos que pueden pagarlo, o aquellos que pueden entenderlo, entonces seguirá subterráneo, implícito, lejos de poder extender su filosofía a una sociedad del conocimiento; un ideal que convive con una contrautopía de base capitalista.

94 A la fecha, y a modo de marco, como administrador de sistemas para una empresa de cable del medio, mantengo regularmente una plataforma de pago que corre sobre Java (libre). La base de datos es Oracle (propietaria), aunque podría ser perfectamente MySQL o cualquier otra *openosource*. Los usuarios son servidos con Apache Tomcat. Los datos se transportan sobre HTTP vía TCP/IP (estándares abiertos), presentados en navegadores Firefox o Chrome (del proyecto libre Chromium).

## **E.5. Abreviaturas empleadas**

- AC: Variable de Investigación. Forma en que accede a la complejidad. Intenta determinar la secuencia que emplean los sujetos entrevistados para tratar obstáculos complejos, del tipo que requieren cambiar todos los patrones empleados hasta el momento para pasar a estrategias radicalmente nuevas.
- AIE: Auxiliar de Informática Educativa
- AN: Variable de investigación. Aptitudes Necesarias. Busca en las entrevistas presencia o ausencia de elementos de razonamiento lógico-matemático o de voluntad suficiente, tales que le permitan al estudiante y en forma independientemente de su vocación, terminar sus estudios.
- API: Interfaz de Programación de Aplicaciones: métodos que publica un programa para permitir a otro programa acceder ciertos datos. El objeto es facilitar a la tarea de intercambio de datos, y especialmente mantener una capa de seguridad que aisle a la base de datos de los requerimientos del exterior. En el contexto que se emplea en esta obra, y ante la carencia de API que poseen los aplicativos web que brinda el gobierno, se recomienda a los encargados de laboratorio que construyan elementos para automatizar la laboriosa carga o descarga de datos (por ejemplo, exportando a lenguaje Ruby, Python, Java, etc, macros creadas en Selenium).
- APK: Formato de paquete instalador para sistema operativo Android. En el contexto del texto utilizado aquí refiere a la creación de un programa para celulares, y dejar disponible su instalador como archivo APK a los alumnos o profesores.
- ATP: Auxiliar de Trabajos Prácticos
- BA: Variable de investigación. Rastrea señales de baja autoestima para enfrentar lo que pudiera encontrarse fuera de la escuela.

- BSD: Trata de una forma de licenciamiento libre y sin restricciones de software. Los programas se acompañan también del código fuente para construir otras versiones derivadas. Quienes publican programas bajo esta licencia, permiten que otras personas lo usen sin restricciones ni "copyright". A diferencia de GPL (ver "GPL"), los usuarios pueden, si desean, no permitir el acceso a las novedades introducidas en los códigos fuentes. Así es la base, por ejemplo, la base del sistema operativo Mac OS/X.
- CAQDAS: Software de Análisis de Datos Cualitativos. Genéricamente, cualquier software capaz de brindar asistencia en investigación cualitativa. Análogo a QDA.
- CATMA: Computer Assisted Textual Markup and Analysis. Software de asistencia para métodos cualitativos, gratuito, mantenido por la Universidad de Hamburgo.
- CLI: interfase por Línea de comandos. Refiere indistintamente a métodos que permiten introducir órdenes a los programas, típicamente sistemas operativos. En el sentido en que se las emplea en la presente obra, refiere a la potencialidad de las CLI para automatizar y crear programas simples que copien archivos, borren y realizan diversas tareas de administración del equipo.
- CM: Componente Motivacional. Variable de investigación empelada en las entrevistas para detectar presencia (o ausencia) de actitudes relacionadas con autoconvencimiento, automotivación y pasión por la carrera elegida. Busca detectar cambios en la vocación elegida.
- CSS: Lenguaje que describe la presentación de datos de páginas XML o HTML
- DPU: Variable de investigación: dificultad en el Pre Universitario o Curso Confrontación Vocacional y Nivelatorio. Indaga en las entrevistas sobre el

esfuerzo puesto por la universidad o institución puesto en ofrecer claramente a los ingresantes, una visión clara de la dificultad que cabe esperar en el ejercicio de la profesión tras egresar.

- IE: Variable de investigación: refiere a la probabilidad que el aspirante inicie o no estudios de sistemas. Puesto que los contenidos ofimáticos y elementales brindados en la escuela secundaria no asoman en ningún momento a la dificultad que cabe encontrarse en el ejercicio de la profesión, se mide presencia de Pre Universitario (PRE) realizado con una completitud o dificultad (DPU) tal que permita al ingresante predecir algo de lo que vendrá después. Algoritmos por ejemplo, es un contenido en los pre universitarios de sistemas que contribuye mucho a sincerar la dificultad de lo que vendrá luego. Se construye esta variable para trabajar la diferencia con otras carreras, por ejemplo Matemáticas, en los que el alumno puede tener una idea más cabal de lo que le espera, como continuación lógica de lo aprendido.
- IRC: Internet Relay Chat: Protocolo de comunicación que da nombre a un conjunto de redes en los cuales se puede encontrar varios canales de charla. En el contexto empleado en la presente obra, refiere específicamente a la red Freenode, una red IRC mantenida por varias universidades europeas, donde se mantienen online cientos de canales sobre ciencia, lenguajes de programación, sistemas operativos, y diversos tópicos relacionados.
- GNU: Se trata de un acrónimo recursivo, que significa "GNU It's Not Unix". Bajo este chiste entre programadores, se esconde un juego de utilidades, librerías y compiladores amparados bajo licencia general pública o GPL. Este software permite la construcción de programas compatibles con Unix, pero sin caer en problemas de licenciamiento. Todas estas utilidades se

han empleado para la creación del sistema operativo conocido como Linux, o "GNU/Linux", y también de BSD (ver "BSD").

- GPL: Se trata de una licencia de software según la cual, al igual que BSD (ver "BSD") los programas se entregan con su código fuente para quien quiera pueda modificarlo y construir nuevas versiones. No obstante estas modificaciones, a diferencia de BSD, deben también quedar disponibles para otras personas.
- MEP: Maestro de Enseñanza Práctica
- PRE: Variable de investigación. Indaga presencia o ausencia de Pre Universitario formal durante el inicio de cursado de las carreras.
- QA: Quality Assurance. Sin traducción literal que indique el sentido correcto. Es un puesto de trabajo en el cual se prueban piezas de software, páginas web o dispositivos. En el ciclo de producción de software, se intenta que no sea el programador quien busque errores, sino personas con capacidades especiales para detectar cambios y errores.
- QDA: Análisis de Datos Cualitativo.
- RE: Variable de investigación. Búsqueda de señales respecto de rechazos sucesivos en empresas.
- RQDA: Programa de asistencia para investigación cualitativa, libre, que funciona en base a una librería escrita en lenguaje R y mediante las librerías para C++ GTK (GNU Tool Kit)
- SEO: Search Engine Optimizer. Conjunto de reglas para mejorar la presencia (*ranking*) en los buscadores.
- TEEP: Variable de investigación. Terminalidad exitosa acompañado de ejercicio de la profesión. Refiere a alumnos que tras haber terminado sus estudios, ejercieron la profesión fuera del ámbito educativo.
- TGS: Teoría General de Sistemas

- VE: Variable de investigación. Detección de ventajas que encuentra de trabajar en escuelas y grado de contentamiento. Busca también medir niveles de hastío, excusas, malestar, oprobio, falta de desafíos, para detectar finalmente si la elección se debe puramente por factores económicos.
- VD: Variable de investigación. Búsqueda de elementos que indiquen vocación docente adquirida a posteriori.

## E.6. Resumen

"Lo único que puede salvarnos de los efectos nocivos de la tecnología, es más tecnología."

(Entrevista a M. Bunge, *La Razón "Cultura"*, Bs. As. LO-XI-85).

Más allá de la asertividad de las exposiciones realizadas durante el presente trabajo, se busca poner en vigencia diversas problemáticas detectadas en las escuelas. Éstas problemáticas refieren a las consecuencias que acarrea la enseñanza de tecnología basada en planes del siglo XX. Por otro lado indagan en la forma en que los informáticos prestan servicio en los laboratorios, omitiendo todo lo aprendido en la facultad, y cayendo en espirales de problemas recurrentes, más propios de usuarios que de profesionales de sistemas.

Si bien la causalidad es un concepto metafísico (Aristóteles; Kant; Jordi Caïs, 1997, pp. 31), durante el texto se intenta llegar a causas fundamentales antes que a señalar obviedades. Estas causas se desarrollan tratando temáticas de cambio y la complejidad, se intenta explicarlas, y se sugieren actividades paliativas y concretas para tratarlas junto a los sujetos afectados que se encuentran actualmente en ejercicio de sus puestos. Se fundamenta asimismo la necesidad de trabajar adecuadamente la Teoría General de Sistemas en las carreras, y finalmente se construyen varias nociones que deberían incluirse en cursos de formación para directivos, respecto de como se debe tratar y aprovechar el personal informático a cargo.

Tras desarrollar estos capítulos, en un conjunto de apéndices se proponen varias formas de rescatar la incómoda letra "N" en "NTIC" y ponerla nuevamente en contexto; las Nuevas Tecnologías de la Información y de la Comunicación, tal como están planteadas, no tiene nada de "nuevas". Preparan a los alumnos sólo

para ser ubicados en oficinas comerciales, en el equivalente digital de la máquina de escribir y de la calculadora de los años 80. La cinta *The Wall* más vigente que nunca, merced al paradójico respeto que tienen profesores y encargados de laboratorio a la tecnología, a las cosas *que se rompen*, y preparan a los alumnos para entrar sin escalas a las oficinas, considerando solo lo que quieren las empresas en lugar de lo que necesita la humanidad.

Esta afirmación, se realiza siguiendo como hilo conductor el desaprovechamiento que se hace de las computadoras, laboratorios simples donde los estudiantes podrían desarrollar los primeros rudimentos de razonamiento científico, tales que provoquen cambios efectivos a escala global. Tras lo cual, en lugar de quedarse en una postura crítica y estéril, se desprenden varias propuestas para el docente que imparte informática: formas de trabajo que provoquen cambios más significativos que aquellos rudimentos de ofimática y web que actualmente imparten.

## **Corolario: Hacia la extinción de Tecnología como asignatura**

En estos días, se plantea seriamente desde el Ministerio de Educación de la Nación nada menos que la extinción del espacio curricular de Tecnología, para descorazonamiento e inquietud de los profesores, quienes planean toda clase de acciones al respecto. En mi posición de docente de la misma área debería unirme a tales acciones; sin embargo admito que estas acciones son sólo de protesta, no de cambio. El estado no quiere mantener una revolución que si bien fue impulsada desde el *hardware* (las *netbooks*), no pudo ser completada, y no por culpa de las maquinas ni del software, sino por factores humanos.

A este paso es normal —y ciertamente justificado— que los docentes de tecnología vayan a ser relegados a meros ayudantes de profesores de otras áreas curriculares. Al fin y al cabo no estaban haciendo otra cosa que impartir clases sofisticadas de mecanografía: formatear textos rapiñados de Internet, sin la mención siquiera de citados de fuentes, preocupados sólo de que los alumnos usen todos los recursos estilísticos posibles, en lograr un *packaging* final vacío, sin producción propia real ni reflexión de los contenidos.

La única novedad, que se ha incorporado a la asignatura, es la hoja de cálculo: un reemplazo de la calculadora, y ni siquiera en sus funciones científicas. Para ilustrar el punto, sólo hay que observar la escasa transversalidad de contenidos con otras materias fuera de Literatura, Historia o Geografía. Aquellas como Matemática, Física e, incluso, Inglés, normalmente quedan relegadas, simplemente porque requieren de alguna otra herramienta distinta de *Microsoft Word*.

A este ritmo es normal que se proponga la extinción de Tecnología; es lo mismo que ocurriría con cualquier departamento de cualquier empresa. En Buenos

Aires es ya un hecho, de acuerdo a Cucuzza (2013), quien comenta: "el Ministerio de Educación de la Ciudad Autónoma de Buenos Aires no tiene previsto incorporar la informática como materia en la llamada Nueva Escuela Secundaria (NES)".

Debido a lo poco que aporta, el profesor queda irrelevante, redundante, y su espacio áulico propio, privado, vacío<sup>95</sup>. Varias propuestas paliativas que buscan recuperar el vértigo por la tecnología que fuera acaso la característica más significativa de décadas pasadas, se tratan en los apéndices de la presente obra, y se sugieren para aquellos que quieran continuarlas en formatos adaptados a las currículas existentes.

95 Cucuzza, Gustavo (2013). *La informática debe ser una materia curricular*. Recuperado de <http://www.lanacion.com.ar/1629448-la-informatica-debe-ser-una-materia-curricular>, [fecha de consulta: 10 de diciembre de 2013]

## Bibliografía

Codificación empleada:

Las citas se emplean de acuerdo a la página 202, capítulo 7.02 "*Books, Reference Books, and Book Chapters*" que figura en la sexta edición del manual "*Publication Manual of the American Psychological Association*". Por ejemplo, esta versión del manual, última a la fecha (22-08-2014), indica poner la ciudad seguida de la editorial, lo cual es un procedimiento inverso al utilizado en clase.

Respecto de las citas de artículos *online*, el manual no menciona la inclusión de la fecha de consulta; a fin de coincidir con la nomenclatura empleada en la materia, y de ubicar temporalmente la producción del artículo, este último dato se incluirá *ex profeso* en las citas.

Se aclara además que se incluye no solo la bibliografía mencionada en el texto, sino también aquella de base, de tal forma que sirva de referencia para aquellos que deseen realizar sus propias conclusiones.

Ambit, M. (2012). *La bombilla que lleva más de 100 años encendida*. Obtenido en <http://www.esencialblog.es/la-bombilla-que-lleva-mas-de-100-anos-encendida>, [fecha de consulta: 9 de marzo de 2014].

Anónimo (s.f.). *América Latina: futura capital del outsourcing de servicios*. Recuperado de <http://www.losrecursoshumanos.com/contenidos/7487-america-latina-futura-capital-del-outsourcing-de-servicios.html>, [consulta el 7 de julio de 2012].

Anónimo (2015). *Open-source software for volunteer computing*. Recuperado de <http://boinc.berkeley.edu>, [consulta el 4 de setiembre de 2016]

- Anónimo (2015). *Choosing BOINC projects*. Recuperado de <http://boinc.berkeley.edu/projects.php>, [consulta el 4 de setiembre de 2016]
- Anónimo (2015). *Help wanted - programming*. Recuperado de <http://boinc.berkeley.edu/trac/wiki/DevProjects>, [consulta el 4 de setiembre de 2016]
- Anónimo (s.f.). *Derecho renueva su plan de estudios por primera vez en 25 años en la UNCuyo*. Recuperado de <http://www.diariouno.com.ar/mendoza/derecho-renueva-su-plan-estudios-primer-vez-25-anos-la-uncuyo-20160521-n793212>, [consulta el 22 de mayo de 2016].
- Anónimo (1999). *Discuss Grift of the Magi*. Recuperado de <http://www.imdb.com/title/tt0701101/>, [fecha de consulta: 20 de marzo de 2014].
- Anónimo, (2012). *Entrevista a Mario Storti*. UNL. Recuperado de <http://paralelizados.com/?p=1065>, [consulta el 1 de agosto de 2012].
- Anónimo (2016). *Jupyter - Open source, interactive data science and scientific computing across over 40 programming languages*. Recuperado de <http://jupyter.org>, [consulta el 25 de agosto de 2016].
- Anónimo (2011). *Los británicos quieren enseñar a programar software en la escuela*. Recuperado de <http://www.losrecursoshumanos.com/contenidos/8262-los-britanicos-quieren-ensenar-a-programar-software-en-la-escuela.html>, [consulta el 20 de junio de 2012].
- Anónimo (2015). *Overclockwise*. Recuperado de <http://theinfosphere.org/Overclockwise>, [fecha de consulta: 20 de marzo de 2014].

Anónimo (s.f.). *Programa "Todos a la Robotica"*. Recuperado de <http://www.diariouno.com.ar/mendoza/derecho-renueva-su-plan-estudios-primera-vez-25-anos-la-uncuyo-20160521-n793212>, [consulta el 15 de agosto de 2016].

Anónimo (2014). *Proyecto Todos a la Robótica*. Recuperado de <http://www.todosalalabotica.ulp.edu.ar/troboticaasp/paginas/Infoprensa.asp> [consulta el 14 de octubre de 2012].

Anónimo (2015). *Software Engineering Code of Ethics and Professional Practice*. Recuperado de <http://www.acm.org/about/se-code>, [fecha de consulta: 21 de marzo de 2014].

Anónimo (s.f.). *Treemaps*. Recuperado de <http://www.top500.org/statistics/treemaps/>, [consulta el 21 de agosto de 2016]

American Psychological Association (2010). *Publication Manual of the American Psychological Association (6th ed.)*. Washington, DC: American Psychological Association.

Argyris, C. (1990). *Overcoming Organizational Defenses*, Nueva York: Prentice-Hall.

Ashby, W. Ross (1960), *Design for a brain; the origin of adaptive behavior*, New York: Wiley.

Ashby, W. (1965<sup>2</sup>), *Proyecto para un cerebro. El origen del comportamiento adaptativo*. Madrid: Tecnos.

Bachelard, G. (2000), *La formación del espíritu científico*. Buenos Aires: Siglo XXI Editores

Barthes, R. (2009), *Mitologías*. Salamanca: Gráficas Varona.

- Basilachis de Gialdino, I et al (2007). *Estrategias de investigación cualitativa*. Buenos Aires: Gedisa.
- Broadfoot, K. J., et al (2008). *A Mosaic of Visions, Daydreams, and Memories: Diverse Inlays of Organizing and Communicating From Around the Globe*. *Management Communication Quarterly*, 22(2), 322-350. Recuperado de <https://doi.org/10.1177/0893318908323574>, [consulta el 7 de noviembre de 2017]
- Bruner, J. (2004). *Realidad mental y mundos posibles. Los actos de la imaginación que dan sentido a la experiencia*. Barcelona: Gedisa.
- Burbules, N, Callister, T. (h) (2001). *Educación: Riesgos y Promesas de las Nuevas Tecnologías de la Información*. Barcelona: Granica.
- Caïs, J. (1997). *Metodología del análisis comparativo*. Barcelona: Centro de Investigaciones Sociológicas.
- Churión, J. (2001<sup>4</sup>). *Economía al Alcance de Todos*. Caracas: Alfadil Ediciones.
- Ciarlante, J (2016). *TEDxPlazadeMulas - Con permiso para hackear*. Recuperado de <http://tedxtalks.ted.com/video.mason/TEDxPlazadeMulas-Juanjo-Ciarlan>, [consulta el 25 de agosto de 2016]
- Ciarlotti, F. (2011). *Ayurveda y Filosofía*. Buenos Aires: Ediciones LEA.
- Copi, M (1973). *Introducción a la lógica*. Buenos Aires: Eudeba.
- Comesaña, J. M. (2001). *Lógica Informal: Falacias y argumentos filosóficos*. Buenos Aires: Eudeba.
- Covey, S (2003). *Los 7 hábitos de las personas altamente efectivas*. Buenos Aires: Paidós.

Cubo de Ceverino, L. et al (2005). *Los textos de la ciencia. Principales clases del discurso académico-científico*. Córdoba: Comunic-Arte.

Cucuzza, G. (2013). *La informática debe ser una materia curricular*, Recuperado de <http://www.lanacion.com.ar/1629448-la-informatica-debe-ser-una-materia-curricular>, [consulta el 10 de diciembre de 2013].

Dans, E (2013). *La informática se incorpora como ciencia al bachillerato inglés*. Recuperado de <http://www.enriquedans.com/2013/01/la-informatica-se-incorpora-como-ciencia-al-bachillerato-ingles.html>, [fecha de consulta: 6 de octubre de 2015].

Dans, E. (2008). *¿Qué tecnología estudian tus hijos en el colegio?*. Recuperado de <http://www.enriquedans.com/2008/01/%C2%BFque-tecnologia-estudian-tus-hijos-en-el-colegio.html>, [consulta 2 de agosto de 2015].

De Bono, E. (2005), *El Pensamiento Lateral*, Buenos Aires: Paidós.

De Regules, S. (2016), *Las Teorías del Caos y la Complejidad*, Barcelona: Batiscafo S.L.

Dergarabedian, C. (2011). *Tecnológicas preparan su artillería y salen de nuevo a la caza del negocio de la tercerización*. Recuperado de <http://tecnologia.iprofesional.com/notas/112755-Tecnologicas-preparan-su-artillera-y-salen-de-nuevo-a-la-caza-del-negocio-de-la-tercerizacin>, [consulta el 30 de julio de 2012].

Dirección General de Escuelas (2006). *La escuela que no le teme a Internet ni a la PC*. Recuperado de <https://web.archive.org/web/20100112151024/http://weblog.mendoza.edu.ar/noticias/archives/013853.html>, [consulta el 25 de agosto de 2016].

- Doyle, A. (2012). *How Often Do People Change Jobs?*. Recuperado de <http://jobsearch.about.com/od/employmentinformation/f/change-jobs.htm>, [consulta el 15 de julio de 2012].
- Dueñas, B. (2008). *Gran Colisionador de Hadrones, impulsado por GNU/Linux*. Recuperado de <http://www.alcancelibre.org/article.php/linux-gestiona-lhc>, [consulta el 5 de julio de 2012].
- Escuela Normal 9-002 "Tomás Godoy Cruz" (2006). Proceso de Construcción del Proyecto Educativo Institucional. Recuperado de <http://www.normal9002mza.edu.ar/downloads/pei2010/PEI2010.doc>, consultado el 3 de julio de 2012.
- Gómez, E (2002). *Mentes Nómades*. Chile: Universidad de Playa Ancha de Ciencias de la Educación.
- Fisch, K.; & McLeod, S. (2012). *Shift Happens*. Recuperado de <http://shifthappens.wikispaces.com> [consulta el 31 de julio de 2012].
- Foerster, H. (2006), *Las Semillas de la Cibernética*. Barcelona: Gedisa.
- Fromm, E. (2007). *La vida auténtica*. Grupo Planeta. Barcelona: Paidós.
- Fontanarrosa, R. (1999), *20 años con Inodoro Pereyra*. Buenos Aires: Ediciones de la Flor.
- Frapiccini, A. (2009). *Software Libre y Ciencia*. Recuperado de [http://www.jornadasdelsur.org.ar/2009/attachments/025\\_ciencia%20y%20linux.pdf](http://www.jornadasdelsur.org.ar/2009/attachments/025_ciencia%20y%20linux.pdf) [consulta el 7 de julio de 2012].
- Gelbort, R (2013), Escuelas usando Software Libre, [https://web.archive.org/web/20130509200912/http://wiki.gleducar.org.ar/index.php/Escuelas\\_usando\\_Software\\_Libre](https://web.archive.org/web/20130509200912/http://wiki.gleducar.org.ar/index.php/Escuelas_usando_Software_Libre), [fecha de consulta: 21 de agosto de 2016].

- Gilles, G. (2004). *The Adventure Of Numbers*. Rhode Island: American Mathematical Society.
- Gullo, A. et al (2013). *New burns and trauma journal celebrating translational research*. Recuperado de <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4978097/#CR11>, [consulta, 7 de noviembre de 2017]
- Heidegger, M. (1993), *El ser y el tiempo*. Santa Fé de Bogotá: Fondo de Cultura Económica.
- Khamash, M. (2010). *Open Source Owns the Web*. Recuperado de <http://jordanopensource.org/blog/open-source-owns-web-infographic>, [consulta el 4 de abril de 2014].
- Kornblit, A. (coord.) (2007<sup>2</sup>). *Metodologías cualitativas en ciencias sociales. Modelos y procedimientos de análisis*. Buenos Aires: Biblos.
- Kuhn, T. S. (1995<sup>11</sup>). *La estructura de las revoluciones científicas*. D. F. México: Fondo de Cultura Económica.
- Kuhn, Thomas S. (1996<sup>2</sup>). *La tensión esencial. Estudios selectos sobre la tradición y el cambio en el ámbito de la ciencia*. D.F. México: Fondo de Cultura Económica.
- Lakatos, I. (2004). *El falsacionismo sofisticado*. Buenos Aires: Eudeba.
- Le Boterf, G. (2001). *La ingeniería de las competencias*. Barcelona: Gestión 2000.
- Livingstone, I. (2011). *Computer science is an essential part of our children's education*. Recuperado de <http://www.independent.co.uk/opinion/comment>

ators/ian-livingstone-computer-science-is-an-essential-part-of-our-childrens-education-6268915.html, [consulta el 15 de junio de 2012].

Livingstone, I. (2012). *Teach children how to write computer programs*. Recuperado de <http://www.theguardian.com/commentisfree/2012/jan/11/teach-children-computer-programmes>, [consulta: 6 de Octubre de 2013].

Gardner, H. (2004). *Mentes Flexibles*. Barcelona: Paidós.

Krashen, S. D. (1985). *The Input Hypothesis: Issues and Implications*. Londres; New York: Longman.

Manuel, F. (2012). *Scientific Linux y Ubuntu, los sistemas operativos empleados en el CERN para hallar el Bosón de Higgs*. Recuperado de <http://www.genbeta.com/sistemas-operativos/scientific-linux-y-ubuntu-los-sistemas-operativos-empleados-en-el-cern-para-hallar-el-boson-de-higgs>, [consulta el 10 de julio de 2012].

McDougall, P. (2008). *Windows Vista, Office 2007 Expelled From British Schools*. [consulta: 3 de agosto de 2012].

McGonigal, K. (2013). *How to make stress your friend*. Recuperado de [http://www.ted.com/talks/kelly\\_mcgonigal\\_how\\_to\\_make\\_stress\\_your\\_friend?language=es](http://www.ted.com/talks/kelly_mcgonigal_how_to_make_stress_your_friend?language=es), [consulta el 17 de junio de 2016].

McGrath, R. (2008). *Open source programming languages for kids*. Recuperado de <http://archive09.linux.com/feature/155203>, [consulta el 3 de agosto de 2012].

McIlroy, M. y Pinson, E. (1978), *Unix Time-Sharing System Forward*. New York: Bell Laboratories.

Ministerio de Trabajo, Empleo y Seguridad Social (2010). *Dinámica del empleo y Rotación de empresas - Primer Trimestre de 2010*. Recuperado de [http://www.trabajo.gov.ar/left/estadisticas/descargas/oede/INF\\_dinamica201001.pdf](http://www.trabajo.gov.ar/left/estadisticas/descargas/oede/INF_dinamica201001.pdf), [consulta el 28 de julio de 2012].

Morin, E. (1994). *Introducción al pensamiento complejo*. Barcelona: Gedisa.

Nielsen, J. (2013). *Teenage Usability - Designing Teen Targeted Websites*. Recuperado de <http://www.nngroup.com/articles/usability-of-websites-for-teenagers>, [consulta: 1 de febrero de 2014].

O'Connor, J. & McDermott, I. (1998). *Introducción al Pensamiento Sistémico*. Barcelona: Urano.

Padilla, L. (2011). *Niño chileno crea sistema para detectar sismos via Twitter*. Recuperado de <http://www.impre.com/la-gente-dice/viewArticle.action?articleId=281474978851802> [consulta: 30 de julio de 2012].

Peña, I. (2013). *De mendigo a programador en cuatro semanas*. Recuperado de <http://www.abcdesevilla.es/tecnologia/informatica-software/20131008/abci-programador-sintecho-nueva-york-201310071649.html>, [consulta, 6 de octubre de 2013].

Perez, D. (2013) *¿Quién usa Linux?*. Recuperado de <http://humanos.uci.cu/2013/06/quien-usa-linux/>. [fecha de consulta: 9 de marzo de 2014]

Pérez Serrano, G. (2003<sup>2</sup>). *Investigación cualitativa. Métodos y técnicas*. Buenos Aires: Docencia.

Pérez, C. (2012). *Alta rotación laboral marca a generación de menores de 35 años*. Recuperado de <http://diario.latercera.com/2012/03/20/01/contenido/t>

[endencias/16-104229-9-alta-rotacion-laboral-marca-a-generacion-de-menores-de-35-anos.shtml](#), [consulta el 16 de julio de 2012].

Podetti, Esteban. (2007). *Yo contra Internet y Google y Wikipedia y todas las cosas esas!* Recuperado de [http://weblogs.clarin.com/podeti/2007/09/11/yo\\_contra\\_internet\\_y\\_google\\_ywikipedia\\_y\\_todas\\_las\\_cosas\\_esas/](http://weblogs.clarin.com/podeti/2007/09/11/yo_contra_internet_y_google_ywikipedia_y_todas_las_cosas_esas/), [consulta el martes 11 de setiembre de 2007].

Raymond, E. (2001). *Como hacer preguntas de manera inteligente*. Recuperado de <http://www.sindominio.net/ayuda/preguntas-inteligentes.html>, [fecha de consulta: 10 de marzo de 2014].

Raymond, E. (2004), *The Art of Unix Programming*. Estados Unidos: Pearson

Raymond, E. (2000), *The Cathedral and the Bazaar: Musings on Linux and Open Source*. Estados Unidos: O'Reilly.

Raynaud de la Ferrière, S. (1971), *Yug, Yoga, Yoghismo. Una Matesis de Psicología*. Mexico: Editorial Diana

Reynoso, C. (2006). *Complejidad y Caos*. Buenos Aires: SB.

Reynoso, C. (2011). *Redes Sociales y Complejidad*. Buenos Aires: SB.

Rickards, J. et al. (2011). *The Next Big Thing*. Miami: Celebrity Press.

Salus, P. (1994), *Quarter Century of Unix*. Estados Unidos: Addison-Wesley.

Sampieri, R.; Collado; C, Lucio, P. (2010), *Metodología de la Investigación*. D.F. México: Mc Graw Hill.

Schuster, F. (2005<sup>3</sup>). *Explicación y predicción. La validez del conocimiento en ciencias sociales*. Buenos Aires: Consejo Latinoamericano de Ciencias Sociales.

Senge, P. (1992), *La Quinta Disciplina*. México: Granica

Sierra Bravo, R. (2003<sup>14</sup>). *Técnicas de Investigación Social. Teoría y ejercicios*. Madrid: Thompson.

Sluckin, W. (1957), *La Cibernética. Cerebros y Máquinas*. Buenos Aires: Galatea - Nueva Visión.

Stallman, R. (2012). *Por qué las escuelas deberían usar exclusivamente software libre*. Recuperado de <http://www.gnu.org/education/edu-schools.es.html>, [consulta el 3 de agosto de 2012].

Stallman, R. (1989). *GNU General Public License*, [https://groups.google.com/forum/#!msg/gnu.announce/m0Jjj\\_64PeQ/8xL1xkVKJb8J](https://groups.google.com/forum/#!msg/gnu.announce/m0Jjj_64PeQ/8xL1xkVKJb8J), [consulta 4 de abril de 2014].

Storti, M. (2012). *PETSc-FEM: A General Purpose, Parallel, Multi-Physics FEM Program*. Recuperado de <http://www.cimec.org.ar/twiki/bin/view/Cimec/PETScFEM>, [consulta el 1 de agosto de 2012].

Taylor, S. & Bogdan, R. (1987<sup>3</sup>). *Introducción a los métodos cualitativos de investigación*. Barcelona: Paidós.

Wainerman, C. & Sautu, R. (comp.) (2004<sup>3</sup>). *La trastienda de la investigación*. Buenos Aires: Lumière.

Watzlawick, P. (1989). *El Lenguaje del Cambio*. Barcelona: Herder.

Watzlawick, P.; Weakland, J.; & Fisch, R. (1992). *Cambio*. Barcelona: Herder.

- Watzlawick, P.; J. Beavin B; & Jackson, D. (2008), *Teoría de la comunicación humana. Interacción, patologías y paradojas*. Barcelona: Herder.
- Weaver, W. (1948). *Science and Complexity*. *American Scientist*, 36: 536-644.
- Wilde, O. (1900). *The Soul of Man under Socialism*. Recuperado de <http://www.gutenberg.org/ebooks/1017>, [fecha de consulta: 22 de agosto de 2016]
- Wilde, O. (1900). *El alma del hombre bajo el socialismo*. Recuperado de <https://docs.google.com/file/d/0B14Synwe1mHzSzNkWi0zTHROcVE/edit>, [fecha de consulta: 22 de agosto de 2016]
- Williams, C. (2011). *Children to be taught to create software*. Recuperado de <http://www.telegraph.co.uk/technology/google/8920278/Children-to-be-taught-to-create-software.html>, [consulta el 18 de junio de 2012].
- Wolton, D. (2000), *Internet, ¿y después?* Barcelona: Gedisa.