

Complejidad y Tecnologías de la Información

Fernando Sáez Vacas

Complejidad y Tecnologías de la Información

Fernando Sáez Vacas

Primera edición: Octubre 2009

Política de derechos de autor: Fernando Sáez Vacas y la Fundación Rogelio Segovia para el Desarrollo de las Telecomunicaciones ponen a disposición del público en general un acuerdo de licencia ColorIURIS AMARILLO que puede aceptarse *on-line* para obtener permisos sobre este libro más amplios a los previstos por la normativa de derechos de autor.

En concreto, previa aceptación del acuerdo de licencia, se autorizará la reproducción, distribución y comunicación pública para uso privado y sin ánimo de lucro, respetando los derechos morales del autor; así mismo se autorizará la transformación siempre y cuando la obra derivada tenga idéntica política de derechos de autor.

Enlace al acuerdo de licencia *on-line*:

<https://www.coloriuris.net/contratos/7153a3df908ac1879d45535f424dac56>

Edita:

Fundación Rogelio Segovia para el
Desarrollo de las Telecomunicaciones
Ciudad Universitaria, s/n
28040-Madrid

Imprime:

E.T.S.I. de Telecomunicación
Universidad Politécnica de Madrid
Ciudad Universitaria, s/n
28040-Madrid

Diseño de cubierta y
maquetación: Rocio Ortega

ISBN (13): 978-84-7402-365-7

ISBN (10): 84-7402-365-3

Depósito Legal: M-42080-2009

Índice

PRÓLOGO	III
ORÍGENES E HISTORIA PERSONAL DE LA ELABORACIÓN DEL LIBRO	XIII
(Presentación de la primera edición de estas Notas de Curso)	
1. Manejar la complejidad.....	XIII
2. Origen y objetivos de estas Notas de Curso.....	XV
3. Arquitectura sociotécnica de la complejidad de las tecnologías..... de la información	XVI
4. Organización del texto.....	XVII
5. Bibliografía personal sobre complejidad.....	XVIII

Bloque I: Generalidades

CAPÍTULO 1: LA VARIEDAD, EN EL SENTIDO CIBERNÉTICO	3
1. Introducción.....	4
2. Variedad.....	4
3. Variedad y sistemas de control.....	6
4. Variedad y observador.....	7
5. La ley de la variedad requerida.....	9
6. Adaptación = Amplificación + Reducción.....	12
7. Demostración de la ley de la variedad.....	15
8. Resumen.....	19
9. Bibliografía.....	20
CAPÍTULO 2: CONCEPTOS RELACIONADOS CON LA COMPLEJIDAD	21
1. Introducción.....	22
2. Caos.....	22
3. Azar.....	30
4. Incertidumbre.....	32
5. Complejidad y Complicación.....	35
6. Complejidad y Desorden.....	36
7. Complejidad y Redundancia.....	37
8. Resumen.....	38
9. Bibliografía.....	39
CAPÍTULO 3: MARCOS CONCEPTUALES	43
1. Introducción.....	44
2. Warren Weaver. Una clasificación de la complejidad.....	45
3. George J. Klir. Complejidad y Jerarquías de sistemas.....	47
Complejidad en los objetos.....	47
Jerarquía epistemológica de sistemas.....	48
Principios generales de la complejidad de sistemas.....	52
4. R. Flood. Una nueva dimensión de la complejidad.....	53
Complejidad y observador.....	54
Complejidad en los objetos.....	55

Interacciones significativas.....	56
Multiplicidad.....	56
No linealidad.....	57
Asimetría.....	58
No holonomicidad.....	59
Una nueva dimensión de la complejidad.....	61
5. F. Sáez Vacas. Modelo de tres niveles de complejidad.....	63
Complejidad. Definición por extensión.....	63
Complejidad en sistemas artificiales.....	64
Modelo de tres niveles.....	66
6. Resumen.....	69
7. Bibliografía.....	70

CAPÍTULO 4: COMPLEJIDAD Y SISTEMAS 73

1. Introducción.....	74
2. De la célula al Sistema Solar.....	74
3. Sistemas y Complejidad.....	75
Complejidad en la ciencia.....	77
Complejidad en el ser humano.....	79
4. Sistemas Sociales: una complejidad distinta.....	81
5. Resumen.....	82
6. Bibliografía.....	83

CAPÍTULO 5: SISTEMAS: VISIÓN ESTRUCTURAL Y VISIÓN FUNCIONAL 85

1. Introducción.....	86
2. Sistemas abiertos y complejidad.....	86
3. Definición de sistema.....	87
4. Interpretación funcional y estructural de los sistemas.....	89
5. El sistema desde el punto de vista estructural.....	91
Límites de un sistema.....	91
Elementos.....	92
Interacciones.....	93
Entorno.....	94
Jerarquía de niveles.....	95
6. El sistema desde el punto de vista funcional.....	97
Entrada y salida.....	97
Realimentación, positiva y negativa.....	98
No linealidades.....	100
Equilibrio y estabilidad.....	101
Adaptación y cambio.....	101
Evolución y emergencia.....	102
Simulación.....	102
7. El sistema como enfoque global.....	103
8. Resumen.....	104
9. Bibliografía.....	105

CAPÍTULO 6: HISTORIA DEL ENFOQUE SISTÉMICO	107
1. Introducción.....	108
2. El enfoque sistémico.....	108
3. Orígenes del pensamiento sistémico.....	110
4. Los sistemas según el M.I.T.....	112
5. La Teoría general de sistemas.....	116
6. Logros del enfoque sistémico.....	118
7. Crítica a la noción de sistema.....	119
8. Resumen.....	121
9. Bibliografía.....	122
CAPÍTULO 7: TRATAMIENTO DE LA COMPLEJIDAD: SIMPLIFICACIÓN	125
1. Introducción.....	126
2. El problema general de la simplificación. George J. Klir.....	126
3. La simplificación, según Gerald Weinberg.....	129
4. Tempus y Hora. Una estrategia de simplificación, según H.A. Simon....	131
5. Un paso más allá de la simplificación. Bell y Newell.....	132
6. Niveles y Abstracciones.....	134
7. Resumen.....	137
8. Bibliografía.....	138
CAPÍTULO 8: LAS JERARQUÍAS	141
1. Introducción.....	142
2. Una aproximación al concepto de jerarquía.....	142
3. La primera definición.....	145
4. Las jerarquías estructurales en la ciencia.....	146
5. Tres nociones distintas, pero relacionadas.....	147
Jerarquía de estratos.....	147
Jerarquía multicapa de toma de decisiones.....	150
Jerarquía de organización.....	151
6. Una formalización matemática.....	152
7. Principios de la realidad jerárquica.....	155
8. Importancia del concepto de jerarquía.....	156
Construcción de complejidad.....	156
Comprensión de complejidad.....	157
Procesamiento de información.....	157
9. Resumen.....	158
10. Bibliografía.....	158

Bloque II: El modelo HxIxO=IO

CAPÍTULO 9: TEXTOS SOBRE COMPLEJIDAD	161
1. Introducción.....	161
2. El conocimiento, oculto detrás de la complejidad.....	161
Subjetividad en la complejidad.....	163
Aspectos de la complejidad.....	167
3. La sobrespecialización, causa y efecto de la complejidad.....	170
4. Necesidad de un nuevo discurso del método.....	172
Los sistemas, un instrumento.....	173
Los modelos y las jerarquías.....	175
5. "Filosofía" e Ingeniería.....	177
6. Tecnología versus Complejidad.....	178
El ordenador, instrumento para manejar complejidad.....	178
Ingeniería, complejidad y economía de medios. El caso del software.	179
7. Bibliografía.....	181
CAPÍTULO 10: UN MARCO UNIVERSAL PARA LA CONSTRUCCIÓN DE MODELOS	183
1. Introducción.....	183
2. Los elementos.....	184
O: el objeto.....	184
I: la interfaz.....	185
H: el observador.....	187
IO: la imagen del objeto.....	188
3. UFM: un marco general para la construcción de modelos.....	189
El proceso.....	189
Significación "dinámica" de H, I, O.....	191
4. Aplicación del U.F.M. al estudio de la complejidad.....	194
Situaciones de complejidad y sus factores componentes.....	195
Construcción de marcos específicos.....	199
5. Resumen.....	201

Bloque III: Tecnologías de la información

CAPÍTULO 11: LAS TECNOLOGÍAS DE LA INFORMACIÓN	203
1. Introducción.....	203
2. ¿Qué son las tecnologías de la información?.....	204
3. Los procesadores de información.....	205
4. Vectores de integración.....	206
5. La complejidad del mundo y la complejidad de las T.I.....	208
6. Las tecnologías de la información y el modelo $H \times I \times O = IO$	211
7. Resumen.....	212
8. Bibliografía.....	213

CAPÍTULO 12: HARDWARE	215
1. Introducción.....	215
2. Hardware - Software.....	217
Máquinas virtuales o el hardware fantasma.....	217
Integración hardware - software.....	219
VLSI.....	220
3. Clasificación de las arquitecturas de ordenadores.....	222
Datos vs. control, una clasificación tradicional.....	222
Otros tipos de taxonomía.....	223
Una ampliación al trabajo de Flynn.....	224
Química, arquitectura y ordenadores.....	227
4. Ordenadores personales y estaciones de trabajo.....	229
Su majestad el pc.....	229
Estaciones de trabajo.....	232
Ordenadores personales vs estaciones de trabajo.....	235
5. RISC, la simplificación del diseño.....	238
6. Paralelismo.....	242
Arquitecturas paralelas.....	243
Problemas paralelos.....	247
7. Resumen.....	250
8. Bibliografía.....	251
CAPÍTULO 13: COMPLEJIDAD Y SISTEMAS DISTRIBUIDOS	255
1. Introducción.....	255
2. Los sistemas distribuidos débilmente acoplados.....	256
3. Categorías de complejidad en sistemas distribuidos.....	257
4. Algunas soluciones.....	259
El sistema Amoeba.....	259
Las llamadas a procedimiento remoto.....	260
El espacio de tuplas de Linda.....	261
5. Resumen.....	262
6. Bibliografía.....	262
CAPÍTULO 14: MEDIDAS DE LA COMPLEJIDAD DEL SOFTWARE	265
1. Introducción.....	265
2. Estudio del software desde el marco HxIxO =IO.....	267
3. Las métricas como instrumentos para abordar la complejidad.....	269
4. Tipos de métricas.....	271
5. Métricas del tamaño del programa.....	272
Número de líneas.....	272
Métricas de Halstead.....	273
6. Estructura y flujo de datos.....	278
Intervalo entre referencias a datos.....	278
Par de uso segmento-global.....	279
Medida Q de Chapin.....	280
7. Estructuras de control del programa.....	285
Número ciclomático.....	285
Extensión de Myers al número ciclomático.....	288

8. Medidas híbridas.....	289
Métrica de Hansen.....	289
Métrica de Oviedo.....	290
9. Resumen.....	292
10. Bibliografía.....	292

CAPÍTULO 15: EL DESARROLLO DEL SOFTWARE 395

1. Introducción.....	295
2. El ciclo de vida.....	296
3. El modelo de desarrollo en cascada.....	297
Definición.....	299
Diseño.....	300
Codificación.....	301
Integración.....	301
Prueba.....	302
Documentación.....	303
4. Los "productos intermedios".....	303
5. Resumen.....	305
6. Bibliografía.....	305

Bloque IV: Anexos

CAPÍTULO 16: INICIATIVA DE DEFENSA ESTRATÉGICA 307

1. Introducción.....	307
2. ¿Qué es la iniciativa de Defensa Estratégica?.....	308
Planteamiento básico.....	308
Diseño inicial de la S.D.I.....	309
Innovación tecnológica en la S.D.I.....	310
Problemas de integración.....	314
Niveles de complejidad en la S.D.I.....	316
3. Software, la piedra angular del proyecto.....	318
Especificación del proyecto.....	318
Arquitectura.....	319
Desarrollo.....	320
Potencia de computación.....	320
Pruebas.....	321
Tolerancia a fallos.....	322
Investigación software.....	323
4. D.L. Parnas y el pesimismo tecnológico.....	324
5. F.P. Brooks, optimismo tecnológico y otras opiniones.....	325
6. Factores de tercer nivel.....	326
7. Conclusiones.....	329

CAPÍTULO 17: AN INTEGRATED FRAMEWORK FOR OFFICE INFORMATION SYSTEMS DESIGN AND MANAGEMENT 331

1. Introduction.....	331
2. Office automation as a set of distinctions.....	332
3. Understand complexity.....	332
Understand your own office.....	334
Apply technology.....	335
New Distinctions in Information Technologies.....	336
4. The office as a system?.....	338
From individual activities to Cooperative Work.....	339
From Cooperative Work to Human Organizations.....	341
5. The Human Side.....	343
6. Users and Producers, Managers and Leaders.....	345
Top down / Bottom up.....	346
Managing complexity.....	347
7. Conclusions.....	347
A non-technical language for managers.....	348
Opening new perspectives.....	349
8. References.....	349

CAPÍTULO 18: A CYBERNETIC THEORY ABOUT COMPUTER INTERFACES AND HUMAN FACTORS WITHIN A FRAMEWORK OF TECHNOLOGICAL INNOVATION 353

1. Introduction.....	353
2. A new theory on office automation.....	353
3. Office systems are human activity systems.....	356
4. The basic structure of the third level of complexity.....	357
5. The complexity of office technology.....	357
6. Adapting complexities: an application of the law of requisite variety.....	360
Organization versus Office Technology.....	360
Individuals versus Office Technology.....	363
7. Conclusions.....	365
8. References.....	366

Prólogo

A primera vista, resulta un tanto extraño reeditar un libro que se editó -como versión 0 de unas Notas de Curso para una asignatura concreta- por primera y única vez en esta Escuela hace ¡15 años! Por las pistas que he podido rastrear mirando documentos y preguntando por aquí y por allá, lo cierto es que aquella fue una edición numéricamente tan escasa, que yo mismo, su autor, ni siquiera tenía un ejemplar. Seguramente se debió a que la asignatura de Cibernética y Teoría de Sistemas, optativa y con pocos alumnos, estaba a punto de desaparecer con el plan de estudios al que pertenecía. Deducimos que si sobraron algunos ejemplares en el almacén del Servicio de Publicaciones debieron acabar destruyéndose para liberar espacio, porque no queda ninguno. En realidad¹, dicha edición se realizó materialmente en 1992 por cuenta del Instituto Tecnológico Bull, financiador de la última etapa de estudio e investigación, dedicada a ordenar y culminar las aportaciones -en forma de fichas, esquemas y apuntes diversos, junto a una vasta selección de bibliografía- elaboradas en años y años de trabajo, reflexión y sesiones docentes del firmante de este prólogo (por cierto, abarrotado de notas a pie de página). Este Instituto llegó a registrar un ISBN de ese mismo texto, que editó electrónicamente pero, debido a la desaparición de dicho Instituto, nunca llegó a publicarse y pasó a ser gestionada por la Escuela.

Por razones que explicaré enseguida, la segunda edición es prácticamente igual a la primera, eso sí, revisada en su totalidad para eliminar erratas, actualizar un poco la terminología y mejorar algún que otro esquema o gráfico. Asimismo, uno de los anexos de la primera edición, que se publicó posteriormente como capítulo en el libro titulado *Estrategia empresarial ante el caos*, Ed. Rialp, 1993, ha sido sustituido por un artículo mío de 1991, *A cybernetic theory about computer interfaces and human factors within a framework of technological innovation* (título que revela circunstancialmente que ya entonces estaba programando mis neuronas para la asignatura que se crearía unos cuantos años después y que ahora estoy impartiendo: Innovación Tecnológica). Lo verdaderamente complicado ha

¹ Las últimas indagaciones realizadas al respecto para reconstruir los hechos relativos a la primera edición de este libro sobre Complejidad parecen demostrar que ésta es finalmente la realidad, y no, como, guiado por recuerdos confusos de la intensa actividad personal y los cambios rápidos organizativos y de toda clase en la época mencionada, escribí recientemente en la bibliografía de otro libro editado por Fundetel, *Domótica: Un enfoque socio-técnico*, 2006, p. 187, donde aquél se referenciaba como una edición de 1992 (Instituto Tecnológico Bull), reimpresso en 1994 por esta Escuela.

sido extraer del 'agujero digital'² la versión informática de los textos originales, tal como quedaron procesados y elaborados para realizar la primera edición, por estar dispersos en soportes de almacenamiento y en formatos y versiones diferentes de software, cuando no perdidos, o ilocalizables. En parte, escanear sobre copias de papel ha sido una tecnología salvadora.

Para cosechar hay que sembrar

Volviendo sobre las palabras iniciales del primer párrafo, la pregunta que cualquiera se hace es si en un universo tecnocientífico de tan rapidísima evolución tiene sentido publicar algo que ya se publicó hace 15 años. Después de meditar la cuestión, mi respuesta ha sido afirmativa. En primer lugar, si nos fijamos en las teorías, modelos y técnicas sobre complejidad expuestos, que constituyen el grueso del libro, los once primeros y los dos últimos capítulos, no solamente siguen plenamente vigentes, por ser básicamente conceptuales y atemporales, sino que el paso del tiempo los ha hecho más fecundos en la práctica -como mi cosecha personal posterior me ha demostrado- porque la complejidad de las infotecnologías ha aumentado increíblemente en esos años y además sus impactos se producen en todos los ámbitos sociales. Sí, la tecnología está rodeada de una complejidad³ siempre creciente, tanto en su esencia, como en su aplicación y consecuencias. Creciente, y por desgracia en gran medida incontrolable, por lo que uno de nuestros objetivos como técnicos debería ser mejorar su grado de controlabilidad, en lo que dependa de nosotros. En cuanto a los cinco capítulos restantes, que pueden verse como ejemplos de aplicación de los modelos de complejidad al estudio del hardware y software, tienen además un valor documental, aunque los datos técnicos hayan sido superados, como es lógico.

²Es el término que utilizamos algunos para referirnos de forma general a los efectos producidos por la desaparición, los cambios o la incompatibilidad de equipos de reproducción o de procesamiento, sistemas operativos, lenguajes de programación, soportes materiales, formatos de datos, aplicaciones y propiedades físicas de los dispositivos de memoria, que hacen que todo lo escrito sobre ellos no haya dejado traza, como si hubiera sido escrito en el viento, salvo que alguien lo haya salvado intencionadamente (F. Sáez Vacas, *Más allá de Internet: la Red Universal Digital*, Ed. Ramón Areces, 2004, p. 231).

³Aunque suene a paradoja, una de las mayores razones para estudiar la complejidad en sentido amplio y uno de sus mejores resultados es reducir al mínimo la complejidad innecesaria, de lo que se deriva que entre los objetivos principales de tal estudio se cuenta el aprender un cierto arte de la simplificación dentro del marco de la complejidad (simplificación y complejificación son procesos complementarios). Muchos técnicos, por querer sacar el máximo partido de las extraordinarias propiedades operativas y funcionales de la tecnología, generan, sin saberlo, complejidades innecesarias mediante soluciones o diseños difíciles de usar o de mantener, a veces vulnerables ante diversos factores no previstos y frecuentemente incompatibles con el tipo de complejidad humana y social. En ocasiones, esa complejidad, más que innecesaria, se convierte en perjudicial.

Si se mira desde el prisma de la docencia, creo que el conjunto del libro constituye asimismo una referencia histórica, en la medida en que ilustra una muestra de los esfuerzos que hacían algunos profesores en esta Escuela durante los años setenta y ochenta para crear asignaturas con contenidos avanzados poco habituales en los programas de estudio de las ingenierías de aquí y de fuera de España. Como se explicó en su primera edición, los textos están explícitamente preparados como soportes didácticos para abordar tema tan amplio y multidisciplinar.

Aprovecho para lamentarme de que el conocimiento del enfoque sistémico y de las técnicas de complejidad, considerada ésta a mi juicio como dimensión central y por tanto insoslayable de las tecnologías de la información, no esté contemplado explícitamente en nuestros actuales planes de estudio ni sea reivindicado por nadie como un componente indiscutible para el proceso de formación de los ingenieros superiores⁴ de telecomunicación y en general de los especialistas en infotecnologías, posicionamiento grave dado que éstas, en comparación con otras

⁴En los tiempos que corren, para ser superiores de verdad, al menos una élite de ellos debería ser capaz de abordar la realidad, dentro de lo que el autor denomina 'realismo tecnologista', a través de una mínima multidisciplinariedad técnica y de un conocimiento suficiente de la complejidad, entendida como marco conceptual para abordar la realidad de manera integral, a ser posible socialmente contextualizada. Incluso, el autor ha planteado la conveniencia de añadir a la actual una nueva titulación de infoingeniero - un ingeniero más sociotécnicamente "especializado", es decir, interdisciplinar- para desarrollar y aplicar estos enfoques cada día más necesarios en estas sociedades nuestras, avanzadas y de complejidad rampante" (véase en su página web <http://www.gsi.dit.upm.es/~fsaez/>, sección **Otros Artículos y Ensayos**, los artículos *Futuros ingenieros híbridos* y *Necesitamos ingenieros sociotécnicos*). Como escribió el famoso matemático y filósofo inglés A. N. Whitehead, de la primera mitad del siglo XX, "toda realidad es unidad compleja". En 1979, el libro-informe al Club de Roma *Aprender, horizonte sin límites*, Ed. Santillana, definió el aprendizaje innovador "como el instrumento para salvar la distancia que media entre la creciente complejidad del mundo y nuestra capacidad para hacerle frente". El poder transformador de la tecnología permite construir nuevas y variadas realidades de progreso, en bastantes casos inevitablemente asociadas -convendría no olvidarlo- a incertidumbre, riesgos y efectos secundarios indeseados, en un entorno sociocultural que expreso así en la página 51 de mi libro *Más allá de Internet: la Red Universal Digital*: "El ser humano (o sea, la humanidad) inventa tecnología, pero no la comprende".

tecnologías, poseen la característica diferencial de transversalidad⁵, por aplicarse a casi todo y como herramienta o soporte indispensable de casi todas las disciplinas, incluidos el estudio y la práctica de las demás tecnologías y ramas científicas. Repasando sus objetivos fundacionales, parece lógico que Fundetel⁶ publique este libro y deje constancia escrita para quien quiera reflexionar sobre este tema.

¿Complexmanía?

A lo mejor, mis convicciones acerca de la importancia de los conocimientos sobre la complejidad son una manía personal, a las que en todo caso he dedicado, sin planificación estudiada y prácticamente sin recursos externos, pero cada día más convenido, un altísimo porcentaje de mis horas de trabajo. Por si esa "manía" tuviera, como creo, una fundamentación lógica y hasta filosófica⁷-emplear y difundir un método de razonar y actuar pascaliano o moriniano, en lugar de cartesiano-, utilizaré,

⁵En general y resumida en forma telegráfica, la tesis que sostengo (véase "Presentación de la primera edición de estas Notas de Curso") es que la complejidad es la dimensión desde la cual puede construirse una arquitectura sociotécnica completa de las tecnologías de la información. J. Rosnay sostiene que la información, la energía y el tiempo son tres sectores fundamentales del conocimiento (*El macroscopio: Hacia una visión global*, Editorial AC, 1977, pag. 5).

⁶Para quien no lo sepa o no lo recuerde, Fundetel es el nombre resumido, o acrónimo, de la FUNdación Rogelio Segovia para el DEsarrollo de las TELEcomunicaciones, que se creó en homenaje al profesor Rogelio Segovia, quien fue precisamente el refundador de la Escuela Técnica Superior de Ingenieros de Telecomunicación de Madrid, a la que puso en línea con los mejores centros académicos del mundo.

⁷L. Winner, en su libro *La ballena y el reactor*, Gedisa, 1987, escribió: Los ingenieros no parecen ser conscientes de las cuestiones filosóficas que su trabajo pueda entrañar (...) Los muy pocos que formulan preguntas importantes acerca de sus profesiones técnicas por lo general son considerados por sus colegas maniáticos, peligrosos y radicales. Por su parte, Mario Bunge, en un artículo publicado en 1991 en la revista TELOS, 24, resalta y analiza las relaciones entre tecnología, ciencia y filosofía, que sintetiza en lo que llama *Sistema Técnica-Ciencia-Filosofía, un triángulo fértil*. Entre otras observaciones, argumenta sobre "cómo el descubrimiento del electromagnetismo supuso también el comienzo de la decadencia de la concepción mecanicista del mundo. Según su visión, otra técnica joven y rica en ideas filosóficas es la que puede llamarse gnoseotécnica, que abarca a la informática y a la inteligencia artificial. Esta técnica, a su vez, se apoya sobre varios pilotes, entre ellos la lógica, el álgebra abstracta, la física del estado sólido y la electrónica. La física del estado sólido no es extraña a la filosofía, se funda sobre la mecánica cuántica, heredera del atomismo antiguo, y se desarrolló entre las dos guerras mundiales en medio de vivas controversias filosóficas. Por si fuera poco, la gnoseotécnica ha replanteado el viejo problema metafísico de la relación entre la mente y el cuerpo".

precisamente en el año de mi jubilación, la oportunidad que me ofrece Fundetel para compartir algunas de mis experiencias profesionales, dejando constancia también en este prólogo de esos esfuerzos continuados y de algunas referencias documentadas (a añadir a las que se recogen en las próximas páginas dedicadas a exponer los "Orígenes e historia personal de la elaboración del libro", en la sección titulada "Bibliografía personal sobre complejidad"), que prueban abrumadoramente la fertilidad aplicativa de tales conocimientos en el Nuevo Entorno Tecnosocial.

Dicha bibliografía personal abarcaba más o menos desde el año 1976 hasta el 1991, por lo que en ella no se citaba un texto anterior, de diciembre de 1973, muy importante para mí por tratarse de mi Memoria para opositar a la Cátedra de Ordenadores Electrónicos de esta Escuela, en el que ya mostraba de modo explícito mi preocupación por emplear enfoques multi- e interdisciplinarios, resaltando junto a la dimensión técnica de la asignatura las dimensiones económica y sociológica⁸, así como la apertura en un futuro a la dimensión ecológica. Creo que dicha Memoria, publicada como libro, mantiene su interés actual por su modo de plantear, diseñar y justificar una asignatura. Entre otras aportaciones, como curiosidad digna de consultarse en los tiempos que corren con aires de Bolonia, expone mi método A.T.E. (Aprendizaje, Test, Enseñanza), prioritariamente centrado en el aprendizaje, que presenté como comunicación en el Seminario Internacional de Formación Profesional en las Telecomunicaciones, organizado por la Unión Internacional de las Telecomunicaciones, Evian (Francia), marzo de 1972.

⁸En cuanto a esta dimensión concreta, la Memoria argumenta ampliamente la conveniencia de ir a diseños sociotécnicos, de los que en su página 71 dice lo siguiente: "Los diseños sociotécnicos, de gran dificultad intrínseca, sólo se generalizarán cuando las personas abocadas a realizarlos, o a tomar decisiones sobre los mismos, estén provistas de un bagaje educativo de índole sociotécnica" (F. Sáez Vacas, *Memoria de Cátedra Grupo XXVII Ordenadores Electrónicos*, ISBN 84-400-7026-8, Dayton, Madrid, 1973, 259 páginas; puede consultarse en la Biblioteca de nuestra Escuela con la referencia de título *Ordenadores electrónicos* y signatura R.0360). Todavía transcurrirían casi 15 años más hasta que empezara a hacerse patente la socialización de la tecnología informática gracias a la difusión de los ordenadores personales, fechas en las que el mismo autor publica su libro *Computadores personales: Hacia un mundo de máquinas informáticas*, Fundesco, 1987, aplicando en él modelos propios de enfoque sistémico y complejidad.

A Gustavo y Jesús: gracias

Como señalé en el primer párrafo de este prólogo, la última y dura fase de ordenación y síntesis de los documentos, fichas y esquemas realizados durante muchos años de trabajo personal, recibió una ayuda económica del Instituto Tecnológico Bull, negociada por el autor, pero lo importante es resaltar que en esa fase colaboraron decisivamente dos alumnos: Gustavo Alonso García y Jesús González Barahona, a quienes, al cabo de tantos años quiero transmitir una vez más mi agradecimiento y manifestarles que su trabajo tiene en esta publicación un efecto duradero. Parte de su colaboración se plasmó además en sendos proyectos de fin de carrera, propuestos y dirigidos por mí en nuestra Escuela. Gustavo, que además firmó conmigo comunicaciones en dos congresos internacionales, defendió su proyecto, titulado *Marco conceptual y modelo sistémico cibernético de la Ofimática*, el 1 de septiembre de 1989. Jesús defendió el suyo, *Estudio de los problemas relacionados con la complejidad de las tecnologías de la información*, el 18 de diciembre de 1990.

Me satisface decir que ambos han orientado su vida profesional a la docencia e investigación universitarias. El primero, después de estudiar y obtener en 1994 un PhD en Computer Science en la Universidad de Santa Bárbara (California) y de ser investigador visitante en el Almaden Research Center de IBM, acabó en el ETH (Instituto Federal Suizo de Tecnología) de Zurich, donde obtuvo el grado de 'full profesor' en 2001. Actualmente trabaja en el Department of Computer Science y dirige el Institute of Pervasive Computing de dicho departamento. Jesús, tras graduarse como doctor ingeniero en nuestra Universidad Politécnica en 1998, trabajó como profesor en la Universidad Carlos III y hoy día es Profesor Titular en el departamento de Sistemas Telemáticos y Computación de la Universidad Rey Juan Carlos.

Una cadena de conceptos, modelos y teorías sobre la Complejidad Sociotécnica de las Tecnologías de la Información. ¿Hacia una tecnocultura?

Los contenidos de este libro se justificaron en su día, aparte de por su interés y actualidad, también por su naturaleza embrionaria o como simiente para futuras cosechas, que podrían incluir frutos tales como perfeccionamiento de sus enfoques, aportación de nuevos elementos o aplicación de sus conceptos, modelos y teorías a aspectos diversos de las tecnologías o de su impacto. Y así ha sido. Esos contenidos le han servido una y otra vez a su autor, desde 1992 hasta la actualidad, para ser aplicados a nuevos y diferentes campos, renovando, reenfocando, incluso en ocasiones ampliando con ellos la visión de la infraestructura tecnológica general, de la domótica, de las redes o de la innovación tecnológica, como podría comprobar sobradamente quien analizase mi bibliografía en este período. Dado que esta comprobación parece poco viable en la práctica, para dar alguna idea de lo que quiero expresar terminaré con unas notas breves sólo sobre algunos otros libros o capítulos de libros que he escrito o en los que he colaborado.

En general, puedo decir que las técnicas de complejidad, el enfoque sistémico y una visión sociotécnica han actuado siempre como guía y contexto de prácticamente toda una cadena de textos, cuya decisión de hacerlos haya dependido sólo de mí, pero para ser breve ahora mencionaré exclusivamente aquellos apartados que llevan el término 'complejidad' en su título. Por ejemplo, mi capítulo *Tecnología de la información, Innovación y Complejidad*, en el libro de autoría colectiva *Estrategia empresarial ante el caos*, Ed. Rialp, 1993. Sobre Ingeniería del Software y aspectos relacionados, incluyendo un capítulo dedicado a la Complejidad del Software, escribí unas notas tituladas *Ingeniería del software: Factores económicos y humanos*, documento de 45 páginas, cuya última revisión data de junio de 2000. Dedicué una parte de ese texto didáctico a enseñarles a nuestros estudiantes cómo, para dirigir o participar en un complejo proyecto de ingeniería del software (trabajo eminentemente intelectual y por tanto humano al 100%), es imprescindible conocer algo más que la pura tecnología. La segunda parte del documento⁹, titulada *El Factor Humano (Peopleware)*, además de adentrarse en la Gestión del Conocimiento, trata cuestiones de Personalidad, Grupo y Liderazgo y diversos Conceptos y modelos teóricos o empíricos de Inteligencia (teoría factorialista, inteligencias múltiples, inteligencia emocional, inteligencia social, entre otros).

Otro ejemplo, el capítulo 19 de mi libro didáctico digital (publicado en 2003 en mi página <http://www.gsi.dit.upm.es/~fsaez/intl/indicecontenidos.html>) se titula *Gestión de la complejidad en la empresa*. El libro *Más allá de Internet: la Red Universal Digital*, publicado por la editorial Ramón Areces en 2004, y elaborado y

⁹Texto incorporado en mi página web:

http://www.gsi.dit.upm.es/~fsaez/intl/ingenieria_del_software_factores_economicos_y_humanos.pdf. En él puede apreciarse cómo una formación en conceptos de complejidad es conveniente para superar muchas situaciones de complejidad con las que nos encontramos en la vida corriente y que pueden no tener nada que ver con la tecnología. Por ejemplo, la gente tiende habitualmente a usar para toda circunstancia una idea simplista de algo tan extraordinariamente complejo como es la inteligencia, que suele asociar con esa medida de la inteligencia llamada Cociente Intelectual, porque desconoce que en realidad existen múltiples y muy diferentes tipos de inteligencia, y por tanto toma decisiones equivocadas en relación con la circunstancia concreta. Otro error de simplismo consiste en identificar el sistema educativo con el sistema escolar, siendo así que, como explicaba hace poco un catedrático de pedagogía, este último es una pequeña parte o componente del sistema educativo, en el que también hay que considerar otros componentes activos, tales como los tipos de trabajo posibles, las oportunidades de promoción laboral, las estructuras empresariales e institucionales de aprendizaje, los horarios, las ofertas culturales, los hábitos de ocio, la vivienda, etcétera. Hay que aprender a distinguir entre la estructura y funcionamiento de los componentes y la estructura y funcionamiento del sistema, que es el conjunto completo de componentes y sus interrelaciones: los árboles y el bosque.

redactado, al igual que el anterior, dentro de mi época como creador de la asignatura de Innovación Tecnológica, pero que en realidad es una sociotecnovisión de la infotecnología, contiene, dentro de la segunda parte dedicada a exponer unas *Lecciones de Economía Digital*, el capítulo *Complejidad*. Más reciente, el libro ya citado *Domótica: Un Enfoque Sociotécnico*, Fundetel, 2006, dedica a la Complejidad su parte más extensa e importante, el Bloque 2, compuesto por 6 capítulos, para aplicar mi modelo de 3 niveles de complejidad, presentado 23 años antes en el Congreso Internacional de Cibernética, a la obtención de un modelo integral de la domótica más moderna. Finalmente, mi colaboración especial en el libro *Web 2.0*, Fundación Orange, 2007, tiene una sección para exponer *El espinoso asunto del control de la complejidad técnica*.

Impulsado por mis actuales circunstancias personales a dejar aquí señales claras de algunas de mis inquietudes, disquisiciones y críticas, anotaré que de un tiempo a esta parte no dejo de preguntarme si con esta trayectoria particular que estoy describiendo, no habré estado, sin saberlo¹⁰, reuniendo y generando elementos conceptuales para construir unos pilotes de tecnocultura, entendida como una cultura madura y responsable de la tecnología. Lo dejo entrever en la sección *Tecnocultura, una "asignatura" pendiente*, del libro *Web 2.0*, en la que subrayo, entre otros factores, cómo la hipertrofia de la especialización de los saberes humanos nos impide conocer la realidad compleja y cómo afrontamos, intelectualmente desarmados, la incorporación a las vidas, quehaceres y formas sociales de tantos ciudadanos de una enorme diversidad de instrumentos digitales: "Parece indiscutible que los técnicos tienden a encerrarse en sus nichos especializados, ignorantes de los saberes técnicos conexos y de las bases conceptuales e históricas que ilustrarían intelectualmente sus conocimientos, mostrándose insensibles (peor que ser ignorantes) al contexto sociológico en el que opera su tecnología particular. Es decir, son técnicos, pero carecen de tecnocultura, que hoy día, dadas las circunstancias, debería constituir un aspecto esencial de la cultura humana y no digamos ya de la cultura de los técnicos. Lógicamente, estas características tecnoculturales, traducidas a la escala práctica que corresponda en cada caso, se propagan al universo de los usuarios, deteriorando socialmente la aplicación de la tecnología digital".

¹⁰¿Hacemos los técnicos filosofía, sin saberlo, como escribió Bunge en el artículo de TELOS antes citado?

Estoy convencido de que a los profesores de ingenierías como la nuestra nos corresponde meditar sobre estas cuestiones y quizá reformular nuestros enfoques y métodos educativos para enriquecerla formación especializada, guiando a nuestros alumnos hacia un aprendizaje en el que los conocimientos estrictamente técnicos se integren estructuralmente en un modelo sistémico, complejo y responsablemente sociotécnico. Mi mensaje: Enseñar tecnología¹¹, por supuesto, pero potenciándola con una buena metodología.

Al lector que desee adquirir una visión más completa de algunas de mis propuestas de modelos, concebidos bajo la perspectiva intelectual de la Complejidad Sociotécnica de las Tecnologías de la Información, le invito a visitar mi página web, empezando por [http:// www. gsi. dit. upm. es/ ~ fsaez/sociotecnologia.html](http://www.gsi.dit.upm.es/~fsaez/sociotecnologia.html). En dicha página encontrará además una selección de artículos, ensayos y columnas guiados por los mismos principios -inevitabilmente sembrados de neologismos¹² del autor-, que podrían completar la visión de los citados conceptos, modelos y teorías, una cadena de la que uno de sus principales eslabones de partida fue en su día este libro que hoy reeditamos.

¹¹Lo que hace este mensaje, al unir tecnología a metodología, es sencillamente recuperar el sentido completo de los conceptos 'ingeniería' y 'tecnología'. Según el D.R.A.E., la primera acepción de este último término es: "Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico"; la cuarta dice: "Conjunto de los instrumentos y procedimientos industriales de un determinado sector o producto". En cuanto a la ingeniería, un diccionario inglés que tengo a mano al redactar este texto la describe así: "The discipline dealing with the art or science of applying scientific knowledge to practical problems". El significado profundo de la practicidad a la que aluden todas las definiciones tiene que ver con la realidad compleja en sus diversos niveles. Por eso, llegando a sus extremos sociales, Bunge escribe: "La responsabilidad moral y social del técnico es muchísimo más pesada que la del investigador científico, porque el primero diseña o controla la fabricación o el funcionamiento de artefactos o de organizaciones que pueden perjudicar a la gente (...)".

¹²Para distinguir los elementos, objetos, seres, fenómenos y conceptos nuevos es inevitable proponer términos nuevos, neologismos, como se explica detalladamente en el libro *Creación Neológica y Nuevas Tecnologías*, coordinado por F. Vilches, Fundación Vodafone, 2006.

(Nota escrita al corregir las pruebas del libro: en estos momentos, una amiga me regala un libro (en español, Ed. Kairós), cuyo autor, Jon Kabat-Zinn, en su Introducción a la Edición del Decimoquinto Aniversario, fechada en septiembre de 2004, dice que "el contenido de su libro no ha sido alterado, que lo único que se ha hecho es añadir esta Introducción". Curiosa coincidencia con las circunstancias del mío, si bien el suyo se dedica a las técnicas de meditación con atención plena y éste a la complejidad. Se me ocurre que bien podría ser que para afrontar emocionalmente los efectos estresantes de la complejidad sociotécnica de las tecnologías de la información hubiera que practicar esas técnicas MBSR (Mindfulness-Based Stress Reduction) de meditación kabatiana. Escribe Kabat en el segundo párrafo de esta Introducción que "si nos transportamos a 1990, cuando este libro apareció por primera vez, notamos que el mundo ha cambiado tremendamente desde entonces, de forma impensable, quizás más que en cualquier época anterior en tan corto espacio de tiempo. Tan sólo observemos el ordenador portátil, los teléfonos móviles, Internet...; el impacto de la revolución digital está prácticamente en todo lo que nos rodea, y la aceleración del ritmo de vida y el nuevo estilo de vida que envuelve los 7 días de la semana y las 24 horas del día (7/24), sin mencionar la inmensidad del cambio social, económico y político que se ha producido de forma global durante este período. La velocidad con la que las cosas cambian en la actualidad es probable que siga, y sus efectos serán más sentidos e inevitables. Se puede decir que la revolución de la ciencia y la tecnología, y su influencia en cómo vivimos nuestras vidas, está apenas empezando. Con certeza, el estrés de adaptación a estos cambios añadido a todo lo demás continuará creciendo en las próximas décadas").

**El autor
Febrero, 2009**

Orígenes e historia personal de la elaboración del libro

(Presentación de la primera edición de estas Notas de Curso)

En abril de 1990 asistí al Workshop **Informatics Curricula for the 90's** (U.S.A.), y allí presenté una comunicación en la que proponía incorporar la dimensión de la complejidad a todas las recomendaciones curriculares relativas a la informática. Pocas fechas antes había aparecido un artículo mío en la revista **Systems Practice**, donde analizaba una experiencia de varios años enseñando a mis alumnos de la **Escuela Técnica Superior de Ingenieros de Telecomunicación de Madrid** a conocer y manejar la complejidad, tal como ésta se manifiesta, no sólo en el ámbito de la informática, sino en el del conjunto de las tecnologías de la información.

Reciente estaba también (diciembre de 1989) la concesión del Premio Fundesco de Ensayo por mi libro **Ofimática Compleja** (publicado en 1990), dedicado a construir (y por tanto a demostrar que se podía construir) una teoría nueva y completa, y, lo que es más importante, una teoría fundamentada en la complejidad, acerca de una de las grandes áreas de aplicación de las tecnologías de la información. El Premio Computerworld, obtenido el mismo año, resultaba asimismo una compensación a la larga época de aventura personal trabajando, en medio de la incomprensión y el silencio, en un campo desconocido.

1. Manejar la complejidad

Al volver de Providence, me detuve un par de días en Boston para visitar la universidad de Harvard y el M.I.T. y dejar transcurrir unas horas interesantísimas en el museo de ordenadores, en cuya tienda adquirí varios libros. Uno de ellos atrajo mi atención más que los demás, porque su objetivo consistía en resaltar la importancia y el auge de la complejidad como un nuevo dominio científico y técnico (H.R.Pagels, **The dreams of reason: The computer and the rise of the sciences of complexity**, Bantam, N.Y., 1989). Respiré contento, porque, aún no satisfaciéndome totalmente su enfoque, el hecho significaba para mí que en el país tecnológicamente más avanzado (por ahora) del mundo empezaba a vislumbrarse a escala más pública una preocupación por la complejidad, considerada como objeto intelectual con entidad propia e independiente de su materialización concreta. Las referencias contenidas en este libro indicaban la emergencia de focos humanos e institucionales interesados en uno u otro aspecto de la complejidad. Venían a sumarse a los que ya me eran más conocidos, singularmente franceses e ingleses.

Pocos meses después, la **Association for Computing Machinery** anunciaba la convocatoria de la primera de una serie de conferencias anuales sobre temas críticos de la informática. Esta se celebraría, como así ha sucedido, y en ella estuve

presente, durante los primeros días de noviembre de este mismo año de 1990 en Arlington (U.S.A.). Uno de los dos temas tratados fue **Managing Complexity**. Aunque tampoco acabó por satisfacerme el enfoque de la Conferencia ni el nivel conceptual de los asistentes, sí lo logró el hecho de que se declarase a la complejidad como un **asunto crítico** y por encima de toda especialidad o aplicación.

No sé si las acciones que se propondrán para el futuro, como resultado de las conclusiones de esta Conferencia, acabarán teniendo alguna influencia en las recomendaciones curriculares según la línea que yo sugerí en Providence meses antes, aunque estoy seguro de que no la tendrán en los planes de estudio españoles que ahora se están diseñando.

Otro hecho digno de ser resaltado es la reciente publicación del prestigioso informe sobre recomendaciones curriculares para la informática preparado conjuntamente (por primera vez) por la Association for Computing Machinery y la **Computer Society** del Institute of Electrical and Electronics Engineering (**Computing Curricula 1991**, ACM Press & IEEE Computer Society Press, dic. 1990). Los requisitos comunes propuestos para todo graduado en informática se dividen ahí en nueve áreas y una complementaria. El área complementaria, dirigida a cuestiones sociales, éticas y profesionales, y otra de las básicas, sobre Interacción Hombre-Máquina, cubren aproximadamente el siete por ciento del total de horas lectivas, y tienen mucho que ver con algunos de los temas que proponemos en este texto. El mismo informe recomienda cursos avanzados sobre Impacto Social de la Informática y sobre Interfaces Hombre-Máquina.

Más significativa es su propuesta explícita de la idea de los **conceptos recurrentes**, definidos como esa clase de conceptos fundamentales que se repiten en todas las áreas y que, por tanto, proporcionan un marco para integrar los contenidos curriculares en los cursos reales. Por su profunda relación con nuestro texto resaltaremos los tres siguientes: la complejidad de problemas grandes, los modelos conceptuales y los niveles de abstracción.

Como podrá comprobar posteriormente el lector, concibo la complejidad como una nueva dimensión desde la cual puede construirse una arquitectura sociotécnica completa de las tecnologías de la información (y no sólo de la informática) y por consiguiente los dos últimos acontecimientos relatados me parecen tímidos y fragmentarios, pero también es verdad que de ellos he recibido el incentivo moral de que había emprendido un camino adecuado con las acciones por mí iniciadas hacía años y que estaban ahora tomando forma didáctica de texto en estas Notas de Curso, cuya redacción acometé a mediados de 1989.

Tal vez sea preciso hacer constar que a nuestro alrededor prevalece una idea generalizada de que las cosas son sencillas, en el fondo. En la vida de cada día se repite una y otra vez la frase "¡es así de simple!". Pocas fechas antes de redactar

esta presentación, y en el curso de un seminario sobre el futuro de la formación de nuestros ingenieros, un ponente, él mismo ingeniero y encumbrado directivo en una empresa de tecnología avanzada, decía textualmente que "las cosas se pueden explicar de manera simple". Creo, por el contrario, que sólo se pueden explicar de manera simple las cosas simples, y que si las cosas complejas se explican de manera simple, o se están trivializando, o se está dando de ellas una versión superficial, o se están mutilando. Mi postura no implica estar a favor de introducir complejidad artificialmente, un arma habitual de técnicos mediocres y de incompetentes, y es compatible con lo que expresa el aforismo anónimo: "hasta lo más pequeño es importante". Según mi experiencia, lo preocupante de las frases anteriores es que expresan un estado de ánimo opuesto al reconocimiento explícito y decidido de la complejidad, como tachándolo de un capricho de algunos que, al parecer, tenemos vocación de complicar las cosas.

2. Origen y objetivos de estas Notas de Curso

Explicaré el origen de estas Notas, aunque no es posible hacer una historia completa, porque, además de aburrir al lector, consumiría mucho espacio. Empecé a interesarme explícitamente por la complejidad aproximadamente cuando en 1977 traduje un libro llamado **El Macroscopio**. Desde 1978 contribuí de forma muy activa a introducir en el plan de estudios de la Escuela de Telecomunicación dos asignaturas tituladas **Cibernética y Teoría de Sistemas I y Cibernética y Teoría de Sistemas II**. Esta última pasó posteriormente a denominarse Ingeniería de Sistemas, y este plan ha continuado formalmente hasta el momento presente. Durante siete años me ocupé personalmente de la primera de estas asignaturas, orientándola hacia una teoría formal de los sistemas, modificando progresivamente, como es lógico, sus contenidos.

Durante el curso académico 1985-86 decidí cambiar radicalmente dichos contenidos y diseñar y construir una asignatura totalmente centrada en la complejidad, utilizando en lo posible, eso sí, el enfoque sistémico. Mi idea de la complejidad incluía, y la incluye todavía con mayor fuerza que entonces, la **complejidad sociotécnica**. La dificultad fundamental para este cambio de eje ha sido siempre la falta de referencias, la ausencia de un mínimo cuerpo de doctrina sobre el que basarse. Como consecuencia natural, he ido inventando, apostando y con mucha frecuencia creando yo mismo los materiales (los modelos, la documentación). Hasta hace muy poco tiempo ni siquiera he podido contar con la ayuda de estudiantes orientados a desarrollar su proyecto de fin de carrera como soporte para mis trabajos, porque la complejidad no era (y en el fondo sigue sin serlo) un tema académico en la Universidad, y menos aún en una Escuela de ingenieros.

El método para mis clases ha tenido que ser improvisado y siempre se ha visto afectado por la servidumbre de tener que sobrellevar la falta de estructuración debida a la naturaleza misma de sus contenidos. En lo que se refiere a la documentación para

su estudio y consulta, en cada curso he propuesto una relación (inconexa a los ojos de cualquier observador) de artículos y capítulos de libros, a veces, y no por egocentrismo, de mi propia y creciente cosecha, como ya se ha dicho.

En todo momento estaba en mi ánimo resolver este problema, que, entre otras cosas, impedía la transferencia de conocimientos fuera de nuestra Escuela y hasta el hecho banal de verme sustituido por causas de enfermedad o viaje, ya que el secreto, por así decirlo, de las conexiones entre los referidos artículos sólo estaba en mis fichas y en mi cabeza. La solución al problema radicaba en confeccionar un libro o unos apuntes, y esto último es lo que finalmente es este texto. Pero la índole enormemente abierta e indefinida del tema, junto con la envergadura del empeño, me ha detenido hasta ahora, en que, con la colaboración de Gustavo Alonso y de Jesús González y una ayuda económica del Instituto Tecnológico Bull se ha acometido el trabajo, encarnado en una versión inicial (versión 0).

3. Arquitectura sociotécnica de la complejidad de las tecnologías de la información

El texto ha tomado como punto de partida la idea global de que dentro de los cambios constantes del mundo en general y de las tecnologías de la información en particular, siempre está presente, y cada día de forma más acusada, percutiente y polifacética, la complejidad. Ello plantea un conjunto de desafíos y dificultades, que requieren ser explícitamente afrontados: **la complejidad es una dimensión abstracta**, pero insoslayable, de las tecnologías de la información.

En la medida en que tal complejidad afecta y es afectada por el sistema social, he adoptado un enfoque muy amplio, intentando cubrir ámbitos teóricos, técnicos y humanos. Resumiendo, mi objetivo final ha consistido en diseñar un primer texto sobre la arquitectura sociotécnica de la complejidad de las tecnologías de la información (con mayor énfasis en la informática, porque ésta es hoy, metodológicamente, el motor dinamizador de las T.T.I.I., o T.I., para hacerlo más breve). Sin embargo, no estoy nada seguro de haber sido totalmente capaz de trasladar a lo escrito el espíritu ingenieril orientado a la resolución de problemas que intento asumir en mis clases.

Quiero subrayar que, precisamente, uno de los aspectos a los que he concedido mayor significación y que mejor muestran el enfoque sociotécnico es el de la **convivencialidad**. Sin embargo, ésta ha sido una de las cuestiones más desatendidas en nuestro texto, aunque no suceda así en mis trabajos publicados sobre esta temática. Por suerte, la preocupación por la convivencialidad se ha ido abriendo camino y hoy se reconoce su importancia bajo distintas denominaciones, como Human Computer Interaction o Computer Human Interaction (H.C.I., o C.H.I.), se publican libros y revistas especializadas, y se celebran congresos de varios tipos. Incluso existen cátedras.

La situación actual contrasta bastante con la que existía hace seis o siete años, cuando en una conferencia internacional de 1983 sobre software me rechazaron una comunicación en la que presentaba mi modelo 5p para el desarrollo profesional de software. Los revisores que examinaron el artículo, animados por una visión especializada y simplista como la que comentábamos arriba, simplemente no lo entendieron. Sólo a partir de 1989, algunos textos conocidos en el campo del software han introducido tímidamente en sus nuevas reediciones algunos de estos planteamientos (Factores humanos, diseño de interfaces hombre-máquina, etc).

4. Organización del texto

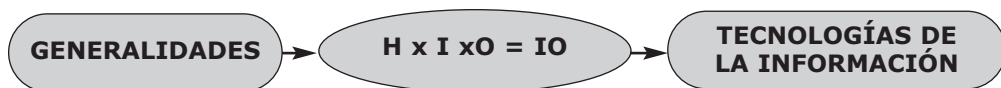
Teóricamente, este texto es un embrión de futuras versiones, en las que tal vez podrá unificarse mejor el estilo, pulir razonamientos, corregir erratas, reducir contradicciones, perfeccionar los enfoques y aportar nuevos elementos o aplicar los conceptos a diversos campos de la tecnología o de su impacto. A propósito no se han eliminado todas las redundancias, por razones de refuerzo didáctico, teniendo sobre todo en cuenta que los contenidos de los capítulos no están necesariamente organizados a la manera secuencial clásica (de hecho, en esta primera versión, ni siquiera se llaman oficialmente capítulos).

En mis fichas y anotaciones personales se acumulan cuestiones y matices que no se han podido recoger aquí, pero ellos seguirán estando en mis clases, como han de estarlo en las clases de todo profesor. Lo que es de esperar es un impacto de la publicación de estas Notas sobre el alumno, porque éste trabajará de manera más adecuada contando con un texto y la bibliografía pertinente. Y a mí, o a quienquiera que emplee este texto como soporte de un curso, me permitirá pasar por alto en clase ciertas descripciones antes obligadas por las carencias de la dispersa documentación anterior.

En todo caso, este conjunto de páginas no tiene otra pretensión, en su estado actual, que servir de soporte a unas clases. Pueden servir también, ¡qué duda cabe!, como lectura autónoma, pero no se garantizan los resultados si el estudiante no dedica un tiempo sustancial a consultar la bibliografía recomendada en cada "capítulo".

Se dividen estas Notas en cuatro partes.

La primera se refiere a **generalidades** sobre la complejidad en sentido abstracto, y sobre teorías, técnicas, y otros conceptos útiles extraídos de la Cibernética y del Enfoque Sistémico.



En segundo lugar, describo un modelo simbólico original (e inédito, porque hasta el presente sólo se ha expuesto en las clases del autor) diseñado para abordar intelectualmente las situaciones de complejidad, y al que he denotado por la fórmula fácil de recordar **H x I x O**. Realmente, es un **marco general para la construcción de modelos** (en inglés U.F.M., por **U**niversal **F**ramework for **M**odelling). Esta parte aparece encabezada por una apretada y variadísima recopilación de textos sobre complejidad, que cubre casi por completo el espectro posible de cuestiones sobre la complejidad, como pretende hacer también el modelo formulado. Cumple además el papel de identificar unas referencias bibliográficas de fondo en el campo teórico y conceptual de la complejidad, que, por su profundo calado, no han sido utilizadas en este texto de orientación didáctica.

La tercera parte se dedica ya a estudiar diferentes aspectos de la complejidad de/en/con algunas parcelas de las **tecnologías de la información**. Es de todo punto evidente que esta parte es la más abierta, puesto que en ella caben otros muchos temas no tratados y hasta diversas formas de tratarlos. Y, en definitiva, los que están lo están principalmente a título de muestra del tipo de enfoque propuesto.

El esquema gráfico adjunto ilustra a lo largo del libro estas tres partes. Lo utilizaremos en formato reducido para ubicar cada uno de los "capítulos", empleando iniciales para simplificar: **G** (por Generalidades), **M** (por Modelo H x I x O) y **T. I.** (por Tecnologías de la Información). Como se verá más adelante, los capítulos de la parte G, que es la más estructurada, usan además otro esquema, que ayuda a seguir en todo momento el desglose de la G en subpartes.

Finalmente, y ya con un formato libre, se recopilan en una sección de **anexos** unos trabajos recientes del firmante y de uno de sus colaboradores. Precisamente, los dos artículos allí recogidos sobre ofimática constituyen un ejemplo práctico del tipo de potente discurso que puede generarse a partir de los conceptos presentados en estas Notas y del sinnúmero de detalles que la consulta de la bibliografía aportará al lector deseoso de profundizar.

5. Bibliografía personal sobre complejidad

Antes hablé de egocentrismo. Me interesa mucho tocar este punto, para rechazarlo, porque sé que al leer este texto algunas personas experimentarán la sensación de que en él hay demasiadas autocitas. La verdad es que sería inadecuado evitarlas, en la medida en que estas Notas se han ido construyendo en una proporción importantísima sobre las aportaciones personales de un buen montón de años. Es decir, que cada trabajo se ha edificado sobre otros anteriores, a los que se hace referencia, si ello se ha considerado pertinente, por no repetirlos.

Pero lo que sucede en realidad es que aspectos muy significativos se quedan fuera del texto y por último desaparecen en artículos o comunicaciones inencontrables, y lo que es seguro es que hasta se pierde traza de los trabajos que uno mismo ha hecho.

Por esto último y por todo lo que vengo diciendo a lo largo de esta presentación, encuentro que, en lo que a mí respecta y pese a engañosas apariencias, estas Notas constituirían una rendición imperfecta e incompleta de cuentas de mi labor sobre la complejidad, si no agrupara en una relación bibliográfica el conjunto de mis trabajos publicados en los que directa o indirectamente se trata explícitamente la complejidad. Que es lo que se hace a continuación.

(Nota: La relación sigue un orden cronológico, aunque se han señalado con un asterisco los libros. El resto aparece indiferenciado: artículos, comunicaciones y otros. Cuando hay más de un autor, ello se indica, aunque sin reseñar su o sus nombres, salvo en el caso de los libros). Es de advertir que durante la corrección de pruebas de estas Notas se han añadido algunas referencias más cuya fecha es posterior a la del término de su redacción. Lo mismo sucede con uno de los artículos del Anexo.

Sistemas, Apuntes dentro del curso Fundamentos y Función de la Ingeniería, E.T.S.I.T., dic. 1976.

Traducción del libro **El macroscopio. Hacia una visión global**, Editorial AC, 1977.

Desarrollo modular y jerarquizado de un sistema software para ayuda al diseño automático de sistemas digitales con estructura modular (coautor), IV Congreso de Informática y Automática, oct. 1979.

Ponencia sobre aplicaciones de la teoría general de sistemas en las ciencias de lo artificial, en Mesa Redonda sobre la Teoría General de Sistemas, 7-8 julio 1980. Publicada en Comunicaciones de las Terceras Jornadas de Automática, Sevilla, mayo 1980, pp. 1-13, con el título Aplicaciones en las Ciencias de lo Artificial.

Contribución al diseño automático de sistemas digitales modulares mediante un sistema software modular y jerárquico (coautor), Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales de Madrid, tomo LXXIV, cuaderno 4, 1980.

Concepción multinivélica y cuasidescomponible de sistemas complejos. Aplicación a la informática (coautor), Comunicación presentada en el V Congreso de Informática y Automática, mayo 1982, Actas pp. 281-286.

Training organization of a middle-size company engaged in information technology services: theory and practice, in: Preprints of IFAC/IFIP Conference "Training for Tomorrow", pp. 187-193, Leiden (Holanda), junio 1983.

El mensaje: invariancia, complejidad, analfabetismo informático y vuelta a los orígenes, Prólogo (con mensaje) a la tercera edición del libro FUNDAMENTOS DE LOS ORDENADORES (coautor).

Facing informatics via three level complexity views, in Proceedings of 10th. International Congress on Cybernetics, Symposium XII: Man in a High Technology Environment (G.E. Lasker, ed.) pp. 30-40. International Association of Cybernetics, Namur (Bélgica), 22-27 agosto 1983.

Las tecnologías de la tercera revolución de la información, Mundo Electrónico, oct. 1983, pp. 133-141.

Propuesta de algunas pautas para guiar la elaboración, a mediados de los ochenta, de los objetivos, metodología y pedagogía de la enseñanza de la informática en cualquier nivel educativo, Ponencia central para un seminario del C.R.E.I. sobre Objetivos, Metodología y Enseñanza de la Informática, nov. 1983. Publicada en el libro Papeles de Buitrago, C.R.E.I., Madrid, abril 1984, pp 10-56. Publicada en la revista Novática, vol. X, num. 55, 1984, pp. 25-39.

El núcleo duro de la informática y los derechos humanos, Lección en el Curso de Informática y Derechos Humanos del Aula Municipal de Cultura del Ayuntamiento de Madrid, Madrid 13-16 marzo 1984. Publicada en la revista BIT, año 6, num.34, 1984.

Some framework ideas for Software Engineering Education, in: Proceedings of International Computer Symposium 1984, Vol. I, pp. 150-156, Taipei (Taiwan), 12-14 dic. 1984.

Cinco subculturas informáticas, Telos, num. 1, 1985.

Convivencialidad, complejidad, computadores e informática, La Vanguardia, domingo, 24-2-1985.

* **Fundamentos de Informática**, libro (coautor, Gregorio Fernández), Alianza Editorial, Madrid, 1987.

* **Computadores personales: Hacia un mundo de máquinas informáticas**, libro, FUNDESCO, Madrid, 1987. Accesit al II Premio Fundesco de Ensayo.

Towards a conceptual remodeling of information technologies based on a broad consideration of complexity, Comunicación en el 31st Annual Meeting of the International Society for General Systems Research, Budapest (Hungría), 1-5 junio 1987.

Ordenadores personales, concurrencia y Quinta Generación, BIT, año 10, num. 50 especial conmemorativo del decenio, 1987.

Tecnologías de la información, ordenadores personales y p.i.p.c., BIT, 11, 56, pp. 37-46, 1988.

Sociedad de la mente y complejidad ofimática, Computerworld, 335,9, 1989, artículo ganador del II Premio periodístico Computerworld España.

Apuntes sobre la percepción social de la informática, TELOS, 16, pp. 13-19, 1989.

Proposal of a three level complexity model for office automation, (coautor, Gustavo Alonso), in Proceedings of the 33rd Annual Meeting of the Int. Society for General Systems Research, Edinburgh, Scotland, jul. 1989.

A complexity architecture for information technologies: a three year didactic experiment, Systems Practice, 3, 1, 1990, pp. 81-96.

Bajo el signo de la trivialización, editorial, TELOS, 20, dic. 1989-febr. 1990, pp. 7-8.

* **Ofimática compleja**, libro, Premio Fundesco de Ensayo 1989, Fundesco, 1990.

Una contribución al estudio de la complejidad ofimática, (coautor, Gustavo Alonso), Novática, XV, 83, 1989, pp. 13-21.

Architectural guidelines for the curricula: 3 layers, 3 new dimensions, 2 basic orientations, different levels in the topics, comunicación presentada en el International Workshop IFIP WG 3.2 on Informatics Curricula for the 1990s, Providence (R.I.), abril 1990.

Subculturas e ideologías informáticas, TELOS, 22, junio-agosto 1990, pp.14-22.

* **Manual de microinformática para directivos**, (coautor, Angel Martínez), Editorial CDN, Madrid, 1990.

La formación del ingeniero de telecomunicación en el año 2000: Reinventar la Escuela, Conferencia escrita para el Seminario sobre EL INGENIERO DE TELECOMUNICACION: HORIZONTE 2000, El Escorial, 27-28 noviembre 1990, publicada en el libro de mismo título, Fundación Universidad-Empresa, Madrid, 1991.

La Sociedad informatizada: Apuntes para una patología de la técnica, Revista de ensayo Claves de Razón Práctica, 10 marzo 1991.

Los derechos humanos y la nueva frontera de la información, PC WEEK, 15 julio, 1991.

Tecnología de la información, innovación y complejidad, V Reunión Internacional sobre Estrategia Empresarial ante el Caos, Seminario Permanente "Empresa y Humanismo", Pamplona, 21 y 22 de Noviembre, 1991.

A cybernetic theory about computer interfaces and human factors within a framework of technological innovation, SISTEMICA, Dic. 1991.

**El autor
Febrero 1991**

Bloque I: Generalidades

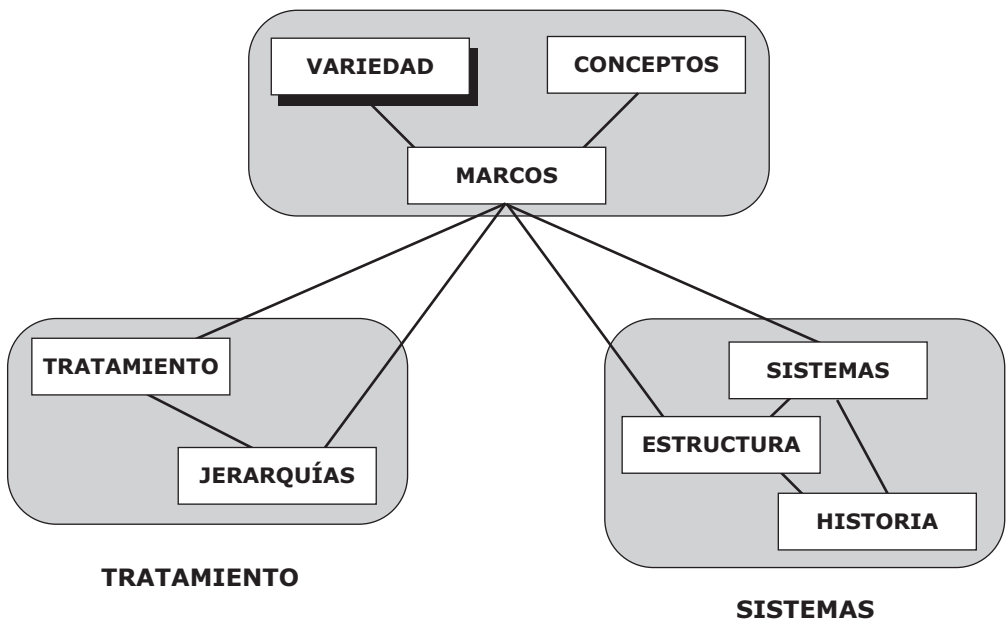
Capítulo 1: La variedad, en el sentido cibernético

Antes de introducirnos de lleno en el estudio de la complejidad, hemos creído conveniente introducir el concepto de variedad, que en general es más intuitivo y sencillo de comprender. Y sin embargo, su potencia es suficiente para que Beer compare la importancia de una de sus consecuencias (la ley de la variedad requerida) con las leyes para la mecánica de Newton.

Aunque el origen de la idea de variedad se encuentra en el ámbito de los sistemas de control, veremos cómo es de gran aplicación en otros muchos campos, incluso en la vida diaria. Nos daremos cuenta, incluso, de las veces que sin saberlo utilizamos en nuestra vida diaria herramientas basadas en este concepto. Probablemente, después de leer este capítulo, seamos capaces de descubrir la variedad y sus aplicaciones por todas partes.



ENFOQUES Y TEORÍAS



1. Introducción

Stafford Beer compara la importancia de la ley de la variedad requerida con las leyes de Newton [Beer, 1974]. En cualquier caso, la importancia del concepto de variedad es muy grande en el campo de los sistemas de regulación, y en muchos otros. Nosotros lo usaremos además como primera toma de contacto con la complejidad, al proponerlo (siguiendo a Beer), como una medida de ésta.

En general, podemos decir que los objetos del mundo que nos rodea son complejos, y por lo tanto tienen asociada una determinada variedad. Nosotros, para comprenderlos y manejarlos, necesitamos absorber de alguna forma esta variedad. Pero las capacidades que tenemos para realizar esto son limitadas: dependen de nuestra propia variedad. Veremos que para poder manejar sistemas de mayor complejidad que la nuestra hemos de usar unas herramientas (físicas o conceptuales): los filtros de variedad.

La variedad, por otra parte, depende en gran medida del observador. Dos personas distintas, con intereses y capacidades diferentes, asociarán con el mismo objeto dos variedades generalmente no iguales. Por eso tiene sentido aplicar el concepto de variedad al conjunto objeto-observador, y no a ninguno de ellos por separado.

La variedad tiene sus leyes, tan inquebrantables como las de Newton (según citaba Beer). De ellas deduciremos el comportamiento de esas herramientas que hemos mencionado: los filtros de variedad.

Para finalizar el capítulo, aportaremos una demostración de la ley de la variedad requerida para un caso que, si bien es restringido, no por ello está exento de interés.

2. Variedad

Comencemos por definir el concepto de variedad, tal y como es entendido por Ashby. Para ello, atendamos a un observador interesado en algún aspecto del mundo que le rodea. Consideremos que el objeto al que dedica su atención está formado por un conjunto de elementos. Pues bien, la variedad de ese conjunto puede ser definida como el número de elementos diferenciables por el observador. Así, el conjunto $\{a,b,a,b,d,a,c,a,b,d\}$ tendrá una variedad igual a 4 (cuatro elementos diferenciables: a, b, c y d)

Intentaremos aclarar el significado de esta definición recurriendo a unos ejemplos, debidos al propio Ashby ([Ashby, 1956], citado en [Sáez Vacas, 1987], pág.115, y [Ashby, 1973], citado en [Klir, 1985], pág.132).

Definamos las clases de cerdos por las propiedades "raza" y "sexo". Un granjero puede distinguir 8 razas distintas de cerdos, pero no es capaz de reconocer si son machos o hembras. Sin embargo su mujer distingue perfectamente los sexos, aunque no entiende nada de razas. Pues bien, ante un conjunto de cerdos donde se den todas las combinaciones posibles de sexos y razas, el granjero percibirá 8 clases diferentes (variedad 8), mientras que su mujer sólo podrá distinguir dos (variedad 2). Si los dos trabajan en equipo, uno encargándose de diferenciar razas y otra sexos, podrán apreciar 16 clases diferentes (variedad 16).

Sea ahora un neurofisiólogo estudiando el cerebro. Sin duda lo considerará como algo extremadamente complejo (con neuronas entrecruzándose, enzimas catalizando reacciones, etc), y distinguirá en él un elevado número de elementos diferentes. Por lo tanto, para el neurofisiólogo la variedad del cerebro es muy elevada. Por el contrario, para un carnicero ese mismo cerebro no es más que un tipo de carne. Los elementos diferentes que distinguirá en él serán muy pocos (quizás incluso lo considere como un único elemento). La variedad que percibirá el carnicero en este caso será muy pequeña.

Una forma también sencilla de entender la idea de variedad es utilizando las unidades de memoria empleadas en los ordenadores. Así, un dígito binario, un bit, tiene dos posibles estados, 1 y 0, su variedad es 2. Un conjunto de ocho bits, un byte, tiene una variedad de $2^8 = 256$. Una memoria de 1 Kbyte posee una variedad de $1.024 \times 256 = 262.144$. De la misma forma, lo que se entiende por una palabra en el argot de los ordenadores (16 bits) tiene una variedad de $2^{16} = 65.536$, mientras que 1 Kpalabra alcanza una variedad de $1.024 \times 65.536 = 67.108.864$.

Aunque a partir de los ejemplos vistos pueda parecer lo contrario, la idea de variedad no suele ser sencilla a la hora de aplicarla a casos no triviales. No siempre se puede obtener una medida concreta de la variedad (así, en el ejemplo del cerebro, ¿cuál sería la variedad que percibe el neurofisiólogo?). Sin embargo es un concepto muy potente, y en muchos casos no es necesario tener una medida exacta, sino simplemente ideas de su orden de magnitud, o de la relación entre las variedades en dos casos distintos, etc. Por ejemplo, volviendo al cerebro, nos ha servido para dar una idea de cómo la imagen que de él tienen un carnicero y un neurofisiólogo no tienen nada que ver (aunque en ninguno de los dos casos hayamos calculado con exactitud la variedad percibida).

3. Variedad y sistemas de control

Beer define la variedad de un sistema como el número de los estados posibles de ese sistema [Beer, 1974]. Y considera a la variedad así definida como una medida de la complejidad de ese sistema. Para él, la complejidad vendría dada por tanto por los estados posibles.

Sea un sistema formado por dos personas, cada una subida en una escalera. Sostienen desde esa altura los extremos de una cuerda elástica, de cuyo punto medio cuelga una pelota. Supongamos que cada persona sólo realiza dos movimientos: subir o bajar su extremo de la cuerda. Mediante estos movimientos, ambos intentan controlar el movimiento de la pelota. Y para darle emoción al asunto, tenemos en la vecindad un gato que gusta de jugar dándole zarpazos de vez en cuando. Esto nos asegura que nuestros sufridos personajes tengan trabajo de manera continua.

Pues bien, ya tenemos el sistema, que además puede quedar definido por un estado de salida: la posición instantánea de la pelota. Además consideraremos que los estados estables (los que nuestros personajes intentan mantener) son aquéllos en los que la pelota está quieta.



Fig. 1. Individuos subidos en escaleras, sosteniendo la pelota

Es fácil imaginar a los dos subiendo y bajando la cuerda, hasta que consiguen mantener la pelota quieta. Si se coordinan bien, es de esperar que lo consigan rápidamente.

Pero supongamos ahora que aparecen más personas que quieren participar en nuestra empresa. Cogen a su vez una escalera cada uno y añaden una nueva cuerda al montaje. Se unen así más elementos (personas y cuerdas) al sistema. Sean así tres, cuatro, cinco... Y el gato sigue incordiando. A poco que nos imaginemos la escena, tendremos a todas las personas gritándose unas a otras, diciendo lo que creen que deben hacer los otros, en un vano intento por coordinarse (supongamos, por ejemplo, que tenemos 20 personajes intentando hablar con cada uno de los otros 19...). La situación puede llegar a ser absolutamente caótica.

¿Qué ha ocurrido? Utilicemos la nueva herramienta que acabamos de introducir:

calculemos la variedad del sistema en cada uno de los casos. Para dos personas, tendremos 4 estados posibles (cada elemento sólo puede tener dos estados: subir o bajar su cuerda): variedad 4. Pero para tres personas serán ya 8 los estados (variedad 8). La variedad crece alarmantemente, según la expresión 2^n , siendo n el número de personas que intervienen.

De este rápido aumento de la variedad se puede deducir (según hemos visto hace un momento) un rápido aumento de la complejidad del sistema, o lo que es lo mismo, de la dificultad para mantener la pelota quieta: el sistema crece exponencialmente en variedad, y así crece también la dificultad para controlarlo. Serán necesarios métodos que reduzcan ésta variedad, si queremos un mínimo de efectividad. Entre éstos pueden estar los siguientes:

- a. Introducción de una jerarquía: cada M personas a las órdenes de un jefe, que limita su libertad (con lo que tendremos menos estados posibles).
- b. Conexiones rígidas entre las cuerdas elásticas, que limitan el efecto de un movimiento.
- c. Obviamente, matar al gato también ayudaría: una vez alcanzado el equilibrio, nos aseguramos de que el sistema se quede en él indefinidamente.

4. Variedad y observador

Algunos de los ejemplos anteriores (en especial el del cerebro y el del granjero) nos muestran de una forma muy directa la dependencia entre variedad y observador. Esta dependencia es tan grande que podemos decir que la variedad no tiene sentido como propiedad intrínseca de los objetos, que sólo adquiere significado cuando se asocia a un observador concreto. Así, hablaremos de "la variedad del cerebro cuando lo estudia un neurofisiólogo", y no de "la variedad del cerebro", o de la de un grupo de cerdos para el granjero (y no simplemente de la variedad de los cerdos).

Algunas veces no hablamos explícitamente del observador, pero lo incluimos de una forma más o menos solapada a la hora de elegir y definir el objeto al que nos referimos. Sería éste el caso del ejemplo de las memorias de ordenador. Implícitamente hemos considerado que el observador está interesado únicamente en el aspecto cuantitativo de almacenamiento de información "bruta" (medida en bits). Si en este caso el observador se hubiese centrado en los aspectos electrónicos (por ejemplo, basándose en la cantidad de componentes necesarios para obtener una pastilla de memoria dada) las variedades obtenidas habrían sido muy diferentes.

Cuando el observador intenta medir la variedad de un conjunto, lo primero que debe hacer (y hace, de una forma más o menos rigurosa), es seleccionar los atributos de los elementos del conjunto que considera relevantes, de acuerdo con sus intereses y posibilidades. Dicho de otra forma, **define** el conjunto con el que va a trabajar. Y es con esta definición con la que está condicionando la variedad que va a medir. En el ejemplo del cerebro, el neurofisiólogo define el cerebro como un conjunto de infinidad de neuronas, y así, la variedad observada será grande. Pero el carnicero lo define como un trozo de carne, con lo que la variedad que perciba será muy pequeña.

Un ejemplo más [Sáez Vacas, 1987]: los esquimales pueden llegar a distinguir más de veinte tipos de blanco en la nieve, lo que "produce" una variedad que es impensable para las personas que no estamos habituadas a vivir en su medio. En este caso, el esquimal tiene la capacidad (dada por su larga experiencia como observador de nieve) y el interés (muchas de sus actividades dependen fuertemente de la nieve) suficientes para definir el conjunto de blancos en función de varios atributos que nosotros no somos capaces de percibir. Para él la variedad del conjunto "colores blancos" es veinte, mientras que nosotros probablemente diferenciamos como mucho entre "blanco mate" y "blanco brillante" (variedad dos).

A la hora de tratar con un objeto, todo observador **selecciona** ciertos atributos. De esta forma, es inevitable que parte de la realidad quede sin considerar. Si el observador es experto y cuenta con unos instrumentos que le den una capacidad de discriminación apropiada, percibirá una variedad mayor que el que sea inexperto en ese campo, o no disponga de los instrumentos adecuados. Apreciará detalles y matices que el otro sólo puede intuir de una forma confusa, o simplemente desconoce. El ejemplo de la percepción del blanco por los esquimales ilustra bastante bien estas ideas. Cuando en la definición de nuestro "conjunto observado" despreciamos algún atributo (por desconocimiento, o porque así lo decidimos para tener un caso más sencillo), ciertos aspectos de la realidad inevitablemente se nos escapan. Así, aunque para nuestro contacto habitual con la nieve nos basta con nuestra idea de "blanco", el detalle con que considera el esquimal esa misma idea le permite realizar deducciones importantes incluso para su supervivencia (por ejemplo, si es capaz de detectar la presencia de una sima escondida bajo una nieve gracias a que presenta un blanco de un tipo especial).

Además, a medida que cambia el punto de vista del observador, aparecen y desaparecen distintas partes del objeto. Así ocurre con los distintos lenguajes de programación. La realidad con la que se enfrentan es, en el fondo, la misma: programar un ordenador. Pero el conjunto de instrucciones, estructuras de datos e incluso conceptos que utilizan varía mucho de uno a otro. Los hay que no permiten manejar bits, sino caracteres, enteros, reales, valores booleanos, o combinaciones de éstos (cadenas, registros, matrices, etc.), como Pascal estándar. Se olvida conscientemente que existe un nivel de trabajo "a base de bits". En cambio, cuando se

programa con un lenguaje ensamblador no se dispone de enteros, ni reales, ni letras, ni valores Verdadero o Falso ni mucho menos matrices, cadenas y tipos de alto nivel. Y esto es así porque los objetivos que persiguen los distintos lenguajes son distintos, tratan con perspectivas distintas del mismo objeto. Imagínese el lector el esfuerzo que supone escribir un mismo programa en Pascal y en ensamblador. Se dirá que depende del programa. Y efectivamente así es. Hacer una base de datos de tamaño medio en un ordenador actual utilizando lenguaje ensamblador pasa por ser una hazaña digna de constar en el Guinness de los Records, mientras que hacer un gestor de interrupciones en Pascal estándar es simplemente imposible. Cada uno de ellos, Pascal y ensamblador, se ocupan de un aspecto diferente de la realidad (el ordenador). Así, el observador, al elegir el lenguaje que utiliza, está eligiendo también los aspectos del ordenador que le interesan. Y definiendo distintos "conjuntos observados", cada uno con su variedad.

5. La ley de la variedad requerida

El padre del concepto de variedad es W. Ross Ashby, quien lo introdujo en un libro ya clásico [Ashby, 1956]. Aparte de proponer el término, Ashby formuló la Ley de la Variedad Requerida (o Ley de la Variedad Necesaria, según otras traducciones), que se ha convertido en ley básica de la cibernética.

Vamos a considerar un sistema cuya estructura está compuesta por dos partes: una que llamaremos **regulador**, y otra, **regulado**. Consideremos que ambas partes pueden tener un cierto número (finito) de estados. En este tipo de sistema, la entidad reguladora recoge información de la regulada, y de acuerdo con esta información, actúa sobre ella. El conjunto de estas acciones y de los estados en los que se encuentran las dos partes dan lugar al comportamiento observable del sistema.

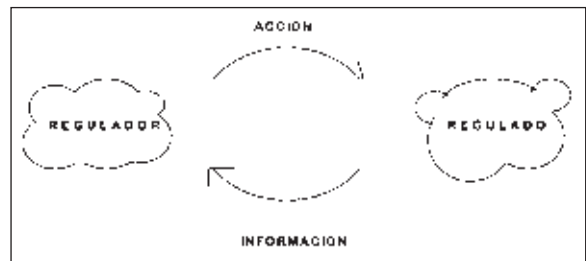


Fig. 2. Esquema básico de un sistema regulador-regulado

Un ejemplo muy sencillo de regulación es el compuesto por el sistema termostato-calefacción. Supongamos una habitación que se desea mantener a una temperatura constante. Para ello instalamos un sistema de calefacción controlado por un termostato. Cuando la temperatura de la habitación esté por debajo de la prefijada, los dispositivos del termostato encenderán la calefacción. Cuando la temperatura deseada se alcance, la apagarán.

En este ejemplo es fácil concluir que el subsistema regulador es el termostato, que ejerce sus acciones (apagar o encender la calefacción), según sea la información que recoge del subsistema regulado (la temperatura de la habitación). El estado estable, por supuesto, se da cuando la temperatura de la habitación es la temperatura objetivo.

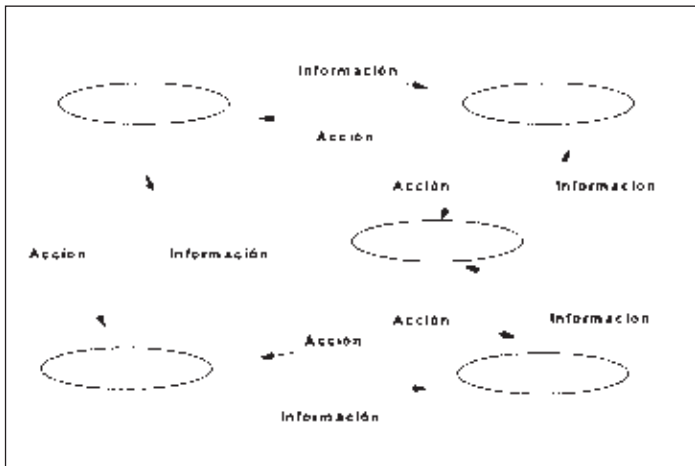


Fig. 3. Varias partes de un sistema regulándose unas a otras.

En problemas reales raramente nos encontraremos con un caso de sistema regulador-regulado puro. Posiblemente la parte "reguladora" esté a su vez regulada por otra u otras, y a su vez regule a más partes que a la "regulada", etc, presentándose por tanto un conjunto muy complicado de relaciones entre los componentes del sistema. Pero nos será muy útil e instructiva

esta simplificación, ya que en ella se dan los fenómenos esenciales de regulación, de una forma suficientemente sencilla como para que podamos sacar una conclusión muy importante: si el subsistema regulador tiene menos variedad que el regulado, no se puede alcanzar la estabilidad. El sistema estará condenado a ser inestable.

Esta afirmación que a primera vista puede parecer demasiado contundente, e incluso infundada, está basada en la ley de la variedad requerida, o ley de Ashby, que exponemos a continuación: **"Sólo la variedad puede absorber variedad"**. Dicho de otra forma: En un sistema regulador-regulado, es preciso que la parte reguladora tenga al menos la misma variedad que la regulada, para que el sistema pueda alcanzar la estabilidad.

Esta ley puede demostrarse matemáticamente de una forma relativamente sencilla, para diversos tipos de sistemas que sean adecuadamente formalizables [Ashby, 1956], por ejemplo, en términos de teoría de la información (ver al final del capítulo, en donde se desarrolla la demostración de la ley de la variedad para un caso sencillo). Pero también tiene una convincente explicación intuitiva: para cada posible estado del regulado, el regulador ha de tomar una acción adecuada si quiere mantener la estabilidad. Por tanto, a cada estado del regulado debe corresponderle

al menos uno en el regulador (que será el que actúa cuando ese estado se produce). Si la variedad del regulador es menor que la del regulado, forzosamente tendrá que haber estados del regulado que no sean controlados (naturalmente, suponemos que los estados están elegidos de forma que sobre cada estado del regulado sólo puede actuar un estado del regulador).

Supongamos un autómata que tiene 256 estados posibles. Si queremos poder manejarlo hemos de desplegar una variedad equivalente. Si sólo tenemos como observadores una variedad de 100, únicamente seremos capaces de abordar 100 estados del autómata, existiendo 156 que quedan totalmente fuera de nuestro control. Traducido a términos más concretos, con una consola con cuatro interruptores, que nos permite una variedad de valor 16, no podremos nunca manejar un robot, pongamos por caso, que tiene 64 posibles estados. Con un volante que sólo permite indicar movimientos a la izquierda o a la derecha no podremos conducir un avión que, al desplazarse en un espacio tridimensional, requiere un control capaz de absorber esa variedad; un mínimo sería un mando que permitiera indicar movimientos hacia arriba, hacia abajo, a la izquierda y a la derecha.

Otro ejemplo muy ilustrativo es el juego de los chinos binarios. Cada jugador muestra su mano cerrada, en la cual puede haber o bien una moneda, o bien ninguna. Cada jugador, por turno, y teniendo en cuenta lo que han dicho los demás y su propia elección, dice un número. Gana quien acierta la cantidad de monedas que esconden entre todos los jugadores.

En este caso, el primer participante introduce una variedad 2 (tiene o no tiene moneda). A medida que aumenta el número de jugadores, aumenta la variedad del conjunto. Así, con dos jugadores la variedad es 3, con cuatro 5, con diez 11, etc. Podremos jugar mientras nuestra perspicacia y capacidad de cálculo sea capaz de absorber esa variedad. Cuando se vean superadas por la variedad introducida por un gran número de jugadores, el juego se convierte prácticamente en un sorteo. En este caso, el sistema regulado lo constituyen el conjunto de totales que se pueden dar (un número de 0 a 9, para el caso de nueve jugadores) mientras que el regulador vendrá representado por cada uno de los participantes (que mediante cálculo y observación intenta abarcar la variedad del conjunto de resultados posibles).

Aquí conviene mencionar la importancia del entorno del objeto y del observador en lo que a la medida de la variedad se refiere. En este ejemplo, sería posible jugar a los chinos con mucha gente si se dispusiera del tiempo necesario para procesar toda la información que se genera. Pero el entorno impone una limitación al problema, en este caso de tiempo. Y ésta es una limitación fundamental, pues no podría desarrollarse el juego si fuera necesario dar una hora para que cada participante pensase la solución (en ese caso, se habría modificado la esencia del juego, con lo que la situación sería distinta). Estas limitaciones son harto frecuentes

y generalmente se consideran asociadas al propio observador, ya que suele ser éste el que establece las condiciones. Pero también pueden verse como imposiciones del entorno, en contraposición a las características e intereses del observador.

De acuerdo con todo esto podemos estudiar un ejemplo de sistema regulador-regulado: una zapatería. Tenemos por un lado la parte que consideraremos reguladora: la tienda entendida como organización. Y por otra la regulada: los clientes. Consideraremos que la variedad que introduce el conjunto de clientes viene dada por sus gustos (en forma de tipos de zapatos que demandan), mientras que la de la zapatería puede calcularse como el número de modelos que oferta. Definiremos además el estado estable del sistema zapatería-clientes como aquel que se da cuando los clientes se quedan satisfechos, habiendo encontrado el tipo de zapatos que buscaban. Aunque no muy rigurosa, esta definición de estabilidad se corresponde bastante bien con la idea intuitiva de que la zapatería se mantendrá mientras los clientes que se acerquen a ella acaben comprando alguno de sus productos.

Pues bien, si sólo tenemos un único modelo de zapatos, no podremos absorber todos los gustos de los clientes: el sistema será inestable (los clientes dejarán de venir). La variedad de los zapatos es 1, mientras que la de los clientes es mucho mayor.

Solamente cuando haya suficientes tipos de zapatos para que cada cliente encuentre uno según su gusto, habremos alcanzado la estabilidad. Ahora las dos variedades están equilibradas.

Por supuesto hay otro caso: que nos dirijamos a un determinado tipo de clientes de variedad 1 (por ejemplo, militares buscando zapatos para su uniforme). En este caso podríamos mantener un único tipo de zapatos sin perder la estabilidad (tal como la hemos definido para este caso). La variedad de las dos partes es en este caso también igual.

La ley de la variedad requerida establece una correspondencia biunívoca entre las variedades de ambos lados de la ecuación. Todos los ejemplos vistos muestran la necesidad de adaptar la variedad entre los dos extremos para que el funcionamiento del conjunto sea correcto. La forma de conseguir esa adaptación es utilizar mecanismos especiales de regulación.

6. Adaptación = Amplificación + Reducción

Visto el significado de la variedad, y con la Ley de la Variedad Requerida en la mano, la pregunta más inmediata es cómo tratar un problema con una variedad distinta a la nuestra. La respuesta a esa pregunta es un concepto muy utilizado en la cibernética: adaptación.

Para explicar mejor la idea de adaptación conviene recurrir al ejemplo de los sistemas de control para luego generalizar en otros campos. En un sistema de control distinguíamos dos partes: el subsistema regulador y el subsistema regulado. Estos van a ser los dos términos de nuestra ecuación de la variedad. Cada uno de estos subsistemas va a tener una variedad determinada, coherente, que no necesariamente equivalente, con la del otro subsistema y relevante a la aplicación del conjunto. La coherencia corre a cargo del observador y es fundamental recordar que no se deben sumar peras y manzanas. La ley de la variedad requerida nos dice que para poder controlar algo tenemos que tener al menos la misma variedad que el objeto a controlar. El ejemplo que veíamos antes del avión y su control es muy ilustrativo en este punto.

En la mayoría de los casos no sucede que el controlador y el controlado tengan una variedad equiparable, ¿qué hacer entonces?. La solución está en la **adaptación de variedades**: reducir la de un lado y aumentar la del otro hasta que se cumpla la ley de la variedad requerida. Y el instrumento que permite realizar esto se denomina **filtro de variedad**. Veámoslo con un ejemplo.

Un profesor (sistema regulador) tiene una variedad muy pequeña frente a la del conjunto de sus alumnos (sistema regulado), desequilibrio que aumenta a medida que hay más alumnos. De una forma harto simplista, pero ilustrativa, podríamos medir la variedad del profesor como el número de horas que puede dedicar a sus alumnos (preparando e impartiendo las clases, resolviendo dudas, evaluándolos...), y la de los alumnos como las que éstos dedicarán a comprender la asignatura (asistiendo a clase, estudiando, etc.). Sería imposible el funcionamiento del conjunto si no hubiera medios para aumentar la variedad del profesor y disminuir la de los alumnos. Imagínese un sistema donde ambas variedades tuvieran que acoplarse directamente: tantas horas de dedicación del profesor, tantas del alumno. Esto corresponde al esquema de clases individuales, con una relación directa entre profesor y alumno.

Pero en la universidad no es éste el esquema elegido: un solo profesor ha de absorber la variedad de muchos alumnos. Para ello, el profesor dispone, por ejemplo, de libros de texto, a través de los cuales los alumnos pueden acceder a información sin necesidad de consultar directamente con él. El libro actúa como un amplificador de la variedad del profesor. La propia organización de las clases, donde un único profesor intenta transmitir su saber simultáneamente a una gran cantidad de alumnos, es otro ejemplo de amplificador de variedad. Para reducir la variedad de los alumnos se utilizan, por ejemplo, los exámenes; el profesor no puede evaluar individualmente a todos los alumnos (de nuevo habría que igualar tiempo de profesor y tiempo de alumno), y por ello establece una serie de preguntas y problemas iguales para todos. El examen actúa como un reductor de la variedad de los alumnos.

Existen otros muchos amplificadores y reductores de variedad relacionados con este mismo ejemplo: horarios de tutoría, bibliografía recomendada, colecciones de problemas, etc. A través de ellos se produce la adaptación de variedad, son lo que se denomina filtros de variedad. Como tales filtros no alteran las características de los sistemas entre los que actúan pero permiten que estos interactúen entre sí. Su funcionamiento es sencillo: proporcionan la variedad necesaria para equilibrar las de ambos subsistemas y por supuesto funcionan en las dos direcciones, es decir, pueden tanto amplificar como reducir, o atenuar, la variedad.

Un **filtro de variedad** es un sistema (o una parte de un sistema) que actúa en dos sentidos. En uno, aumenta la variedad, mientras que en el otro la disminuye. Correctamente aplicado, el regulador ve disminuir la variedad del regulado en la medida necesaria para poder absorberla, y el regulado ve aumentada la del regulador en la misma cantidad.

Naturalmente, no basta con aumentar/reducir la variedad: el filtro debe hacerlo de la forma adecuada, actuando sobre los elementos necesarios para alcanzar el equilibrio.

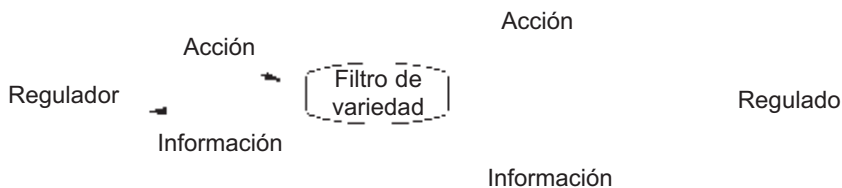


Fig. 4. Filtro de variedad colocado entre un regulador y un regulado, para adaptar sus variedades

Son innumerables los ejemplos de filtros de variedad y estamos tan acostumbrados a verlos, aunque no como tales, que pueden parecer absolutamente triviales aunque en muchos casos están muy lejos de serlo. El ordenador puede ser considerado como un filtro de variedad. Supongamos que lo estamos usando para mantener una base de datos. Por un lado trata con una buena cantidad de información, que seguramente una persona no podría asimilar (sistema regulado). Por la otra, estamos nosotros (sistema regulador), consultando la base, buscando elementos relacionados y pidiendo estadísticas. El ordenador es aquí utilizado para acoplar la variedad del problema que estamos tratando a la variedad de nuestro cerebro (que es la que podemos usar en nuestros análisis).

Las diferentes modalidades de declaración de la renta actúan como filtro entre la variedad enorme de la sociedad (sistema que se intenta regular) y la del ministerio de Hacienda (que actúa como regulador), mucho menor. Un horario de trabajo en una fábrica reduce enormemente la variedad de todos sus trabajadores obligándolos a adaptarse al él y al mismo tiempo amplifica la variedad de los que la dirigen

que pueden así controlar la actividad que se desarrolla. La gama de colores y modelos que ofrece una tienda de ropa permite que ésta amplifique su variedad para adaptarla a la de los clientes, que suele ser enorme (recordar el ejemplo de la zapatería). En sentido inverso, esa misma gama de colores y modelos reduce la variedad de los clientes, que se tienen que ajustar a ella. La moda también se puede interpretar de esta forma. Sin semejante filtro, los productores se podrían encontrar con exigencias de clientes que no pueden satisfacer pues no pueden adaptarse a cada individuo. A través de la moda, se produce una canalización de la demanda que facilita la adaptación de variedad. Esto mismo es aplicable a la oferta de modelos de ordenadores de una determinada casa, a los anuncios, a los libros, etc.

Los conceptos de variedad y adaptación, junto con la ley de la variedad requerida, son unas herramientas intelectuales muy potentes para el análisis de la interacción entre dos entidades. A pesar de su aparente simplicidad, su grado de validez es prácticamente universal y constituyen una de las bases del pensamiento cibernético. Pero no deben considerarse de forma aislada pues, como iremos viendo, hay otros muchos factores en juego.

Que su aplicabilidad es grande se demuestra por las numerosas referencias que se le han dedicado en todo el mundo. Podemos citar esfuerzos muy próximos a nosotros, como es la modelización, a partir de la ley de la variedad, del mercado -aparentemente caótico a mediados de los ochenta- de los ordenadores personales [Sáez-Vacas, 1987] o, por otro, el desarrollo de un punto de vista teórico para los procesos de innovación tecnológica en una empresa [Sáez-Vacas, 1990]. Un campo en el que el concepto de variedad ha conseguido un desarrollo muy especial, aunque todavía pionero en cuanto a su implantación, es en el de la organización de empresas. La figura indiscutible en este área es Stafford Beer, quien, a través de muchos años de investigación, ha cuajado un enfoque de gran nivel para el diseño o el diagnóstico de organizaciones, conocido como el **Modelo del Sistema Viable** (VSM, por sus siglas en inglés) [Beer, 1985]. Muy recientemente, la revista Systems Practice le ha dedicado un número completo [Espejo, 1990].

7. Demostración de la ley de la variedad

Aproximación a una demostración de la ley de la variedad requerida, tomada de [Ashby, 1956, cap. 11]. Supongamos que tenemos dos jugadores, llamémosles Regulador y Perturbador. Las reglas del juego en que están enfrascados son muy sencillas:

- a. Tienen ante ellos una tabla como la de la figura (tabla 1).
- b. Perturbador juega primero, eligiendo un número, al que le corresponderá una fila de la tabla.

- c. Regulador, conocido ese número, seleccionará una letra mayúscula (y por tanto, una columna).
- d. Regulador gana si la letra minúscula que ocupa el lugar donde se cruzan la columna que él ha elegido y la fila que eligió Regulador es una a. Si no, gana Perturbador.

REGULADOR			
	A	B	C
1	b	a	c
2	a	c	b
3	c	b	a
PERTURBADOR			

Tabla 1. Primer juego entre Regulador y Perturbador

Para esta tabla en particular vemos que Regulador puede ganar siempre. Elija lo que elija Perturbador, siempre podrá seguir la siguiente estrategia:

1	2	3
B	A	C

Esto es, si Perturbador elige 1, él puede elegir B, si elige 2, ganará escogiendo A, y si 3, bastará con que seleccione C. De esta forma, el resultado siempre será a.

Si analizamos un poco más la tabla que estamos usando, veremos que Regulador también puede ganar siempre en el caso de que el resultado deba ser b o c. Puede decirse que Regulador tiene un completo control del resultado (que a partir de ahora llamaremos **salida**).

REGULADOR				
	A	B	C	D
1	b	d	a	a
2	a	d	a	d
3	d	a	a	a
4	d	b	a	b
5	d	a	b	d
PERTURBADOR				

Tabla 2. Segundo juego entre Regulador y Perturbador

Pero no todas las tablas posibles van a ser tan favorecedoras para Regulador como ésta. Por ejemplo, supongamos que nuestros dos jugadores deciden cambiar a la tabla 2 (ver figura). Mientras el objetivo sea a, no cambia nada: Regulador puede ganar siempre. Pero si el objetivo es b hay ciertos casos en los que perderá. Por ejemplo, cuando Perturbador elija una de las filas 2 o 3. Y éste no es el peor caso para Regulador: si el objetivo es c, perderá siempre, independientemente de lo que elija su contrincante.

REGULADOR			
	A	B	C
1	f	f	k
2	k	e	f
3	m	k	a
4	b	b	b
5	c	q	c
6	h	h	m
7	j	d	d
8	a	p	j
9	l	n	h
PERTURBADOR			

Tabla 3. Tercer juego entre Regulador y Perturbador

Intentemos sacar alguna conclusión sobre las posibilidades que tiene Regulador de ganar para una tabla dada. En primer lugar, vamos a limitar un poco el tipo de tabla con que se juega. Para ello, impondremos la condición de que las letras no se repitan en la misma columna. Así evitamos que Regulador pueda no cambiar su elección cuando Perturbador cambia la suya. El problema que nos queda es aún suficientemente general, y de gran interés en teoría de regulación.

Así pues, en las tablas que consideramos, Regulador necesitará conocer exactamente cuál es la elección de su contrincante para poder ganar (cualquier cambio de Perturbador habrá de seguirse necesariamente de un cambio de Regulador).

De este tipo es la tabla 3 (en figura adjunta). La estrategia de Regulador puede expresarse como una transformación univaluada (esto es, a cada movimiento de Perturbador le corresponderá uno y solo uno de Regulador). Una estrategia posible podría ser:

1	2	3	4	5	6	7	8	9
C	A	B	A	C	A	B	C	A

De esta forma estamos especificando un conjunto de salidas, las que forzará Regulador para cualquier movimiento de Perturbador. De otra forma,

(1,C)	(2,A)	(3,B)	...	(9,A)
k	k	k	...	l

Es sencillo comprobar que la variedad de este conjunto de salidas $\{k,k,k,\dots,l\}$ no puede ser menor que

$$\text{(Variedad de Perturbador)} / \text{(Variedad de Regulador)}$$

Probémoslo. Para ello, hagamos que Regulador, eligiendo su estrategia, intente mantener la variedad del conjunto de salidas tan pequeño como le sea posible. Para ello, elegirá un elemento de la primera fila. En la segunda fila, debe cambiar de columna si quiere intentar no añadir un elemento distinto al conjunto (recordad que en una misma columna no se pueden repetir elementos). Para mantener la variedad de la salida al mínimo, debe elegir una nueva columna en cada fila. Así, si Regulador dispone de n columnas (3 en la tabla 3), después de la columna n -ésima, debe necesariamente elegir una columna ya elegida, con lo que obligatoriamente estará añadiendo un nuevo elemento al conjunto de salidas. En general, si Regulador dispone de r movimientos diferentes, y Perturbador de p , la variedad del conjunto de salida no podrá ser menor que p/r .

Este juego puede ser considerado de una forma un poco diferente: si Regulador siempre elige la misma columna, sea cual sea la elección de Perturbador, este último ejerce control total sobre las salidas: la variedad del conjunto de salidas será tan grande como lo sea la variedad de los movimientos de Perturbador. En general, si Regulador dispone de r columnas, la variedad del conjunto de salida puede reducirse a $1/n$ de la de Perturbador, pero no a menos. Por tanto, solo la variedad en las posibilidades de Regulador puede disminuir la variedad en la salida. Esto nos lleva directamente a la ley de la variedad requerida: "Sólo la variedad en el Regulador puede disminuir la variedad debida a Perturbador, sólo la variedad puede absorber variedad." El llamado Perturbador es el sistema Regulado.

8. Resumen

Hemos comenzando este capítulo definiendo el importante concepto de **variedad**, según la entendía Ashby, aplicada fundamentalmente a conjuntos. Luego nos hemos acercado a la utilización que Beer hace de este concepto, aplicándolo a los sistemas de control. Es importante remarcar el hecho de que Beer considera a la variedad como una **medida de la complejidad** de un sistema. En los siguientes capítulos profundizaremos en el significado de la complejidad, y podremos comparar esta idea con las de otros autores.

La importancia del **observador** en el concepto de variedad ha sido considerada como fundamental, ya que interviene directamente a la hora de elegir el conjunto con el que trabajaremos (lo que influye claramente en la variedad que se nos presentará). Y en concreto, serán los **conocimientos, intereses e instrumentos** del observador los que más influyan en esta elección. Esta importancia llega hasta tal punto, que cambios de punto de vista del observador producirán la aparición y desaparición de partes de nuestra percepción del objeto.

La introducción de los sistemas de regulación nos permitió la formulación de la **ley de la variedad requerida**, de gran importancia en la cibernética, e incluso en la vida diaria, como vimos en los ejemplos. A partir de estas ideas llegamos al concepto de **filtro de variedad**, herramienta que nos permite manejar sistemas que de otra forma serían incontrolables.

Por último, vimos una de las demostraciones de la ley de la variedad requerida que aporta Ashby, para un caso sencillo, pero no carente de interés, ya que representa bastante bien a los sistemas sometidos a perturbaciones, muy extendidos en la realidad.

El estudio que hemos hecho hasta aquí del concepto de variedad nos permitirá adentrarnos con más facilidad (y ya más "en caliente") en el complicado mundo de la complejidad, que trataremos en los siguientes capítulos.

9. Bibliografía

Tenemos dos apartados. En Notas Bibliográficas se registran las referencias consideradas especialmente valiosas para el estudio del tema. En Referencias Bibliográficas se incluyen todas las referencias citadas en el capítulo.

Notas bibliográficas

El libro de Beer, "**Designing Freedom**" [Beer, 1974] es muy apropiado para tomar idea del impacto producido por la aplicación del concepto de variedad a los sistemas sociales. Además (especialmente en los capítulos 1, 2 y 3), realiza una buena introducción de la variedad, y de algunas de sus consecuencias, como la ley de la variedad requerida y los filtros de la variedad.

En el libro de Ashby [Ashby, 1956] podemos encontrar los primeros estudios sobre el concepto de variedad, la ley de la variedad requerida, etc. Sin duda, este libro se ha convertido en un clásico del tema.

En el artículo de Klir, "**Complexity: some general observations**" [Klir, 1985] encontramos, antes de entrar a tratar con más profundidad otros aspectos de la complejidad, unas cuantas útiles nociones sobre variedad, incidiendo especialmente sobre la importancia del observador (acompañadas de unos cuantos ejemplos, algunos tomados de Ashby, parte de los cuales hemos reproducido aquí).

Referencias bibliográficas

Ashby, W.R. (1956): "**An introduction to cybernetics**", Chapman and Hall, Londres [en español (1972): "Introducción a la cibernética", Ediciones Nueva Visión, Buenos Aires]

Ashby, W.R. (1973): "Some peculiarities of complex systems", **Cybernet. Med.**, num.9, pp.1-7.

Beer, S. (1974): "**Designing Freedom**", Wiley & Sons, Londres.

Beer, S. (1985): "**Diagnosing the System for Organizations**", Wiley & Sons, Chichester.

Espejo, R. ed. (1990): "Special issue: Stafford Beer's Viable System Model", **Systems Practice**, Vol. 3, num. 3.

Klir, G.J. (1985): "Complexity: some general observations", **Systems Research**, vol.2, num.2, pp.131-140.

Sáez Vacas, F. (1987): "**Computadores personales**", FUNDESCO, Madrid.

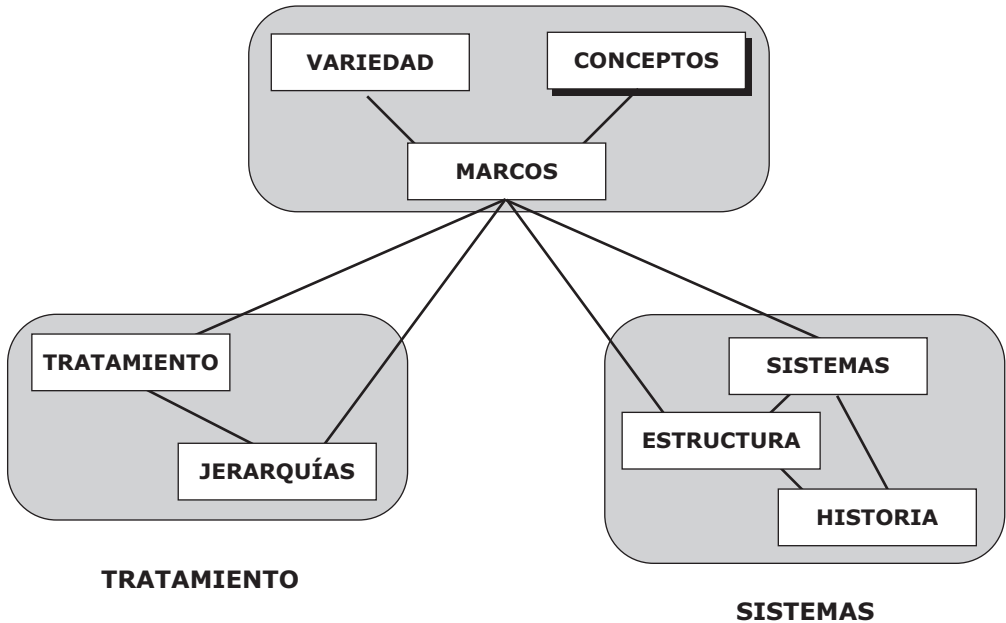
Sáez Vacas, F. (1990): "**Ofimática compleja**", FUNDESCO, Madrid.

Capítulo 2: Conceptos relacionados con la complejidad

El estudio de la complejidad comenzó hace ya algunos años pero sólo recientemente se han empezado a investigar ciertos fenómenos directamente relacionados con ella. La ciencia, tradicionalmente apegada a las linealidades y al determinismo, no parecía aceptar la existencia de comportamientos capaces de refutar muchas teorías establecidas. El Azar, por ejemplo, se ha considerado a menudo como una forma de referirse a lo que no podemos explicar, no como algo intrínseco a la propia Naturaleza. Relacionados con la complejidad, existen muchos conceptos como el caos, el azar, la redundancia o la complicación, a los que se está prestando creciente atención en campos tan diversos como la teoría de circuitos o la meteorología. Muchas de las conclusiones a las que se llega son las mismas que se deducen del estudio de la complejidad.



ENFOQUES Y TEORÍAS



1. Introducción

La complejidad es un término genérico que como ya hemos visto es difícil de definir. Sin embargo hay multitud de otros conceptos que nos son más cercanos y más intuitivos y nos pueden ayudar a tener una idea más clara de lo que es realmente la complejidad.

Ya hemos tratado las fuentes de complejidad. En este capítulo vamos a tratar de desarrollar algunas ideas sobre una serie de conceptos que están muy relacionados con ella pero que deben distinguirse como tales conceptos para evitar confusiones. Por otro lado no hay que olvidar que intentamos estudiar la complejidad para poder abordarla mejor y para ello no hay medio mejor que conocer de donde procede y las características más destacadas de los factores que la producen.

2. Caos

La ciencia tradicional se basa en una serie de aproximaciones que por lo extendidas aparecen como características propias de la Naturaleza en lugar de lo que realmente son, suposiciones que nos permiten modelar el comportamiento de la naturaleza para poder estudiarlo y reproducirlo bajo condiciones controladas y conocidas. Una de estas suposiciones es el comportamiento lineal de los parámetros y de las características relevantes de los procesos naturales. La máxima "iguales causas producen iguales efectos" es una idea extendida basada en la necesidad de un determinismo científico. Sin embargo esa idea encierra varias suposiciones cuestionables.

La primera de ellas y más importante es el término "iguales" que supone que somos capaces de establecer las mismas condiciones, exactamente las mismas, para poder obtener los mismos efectos. Hablar de "iguales causas" es lo mismo que poder establecer las condiciones iniciales con precisión infinita, algo que no es posible. Más adelante volveremos sobre este punto cuando tratemos del azar, aquí lo que nos interesa es la combinación de esa falta de precisión con las no linealidades. La no linealidad es una de las fuentes clásicas de complejidad (Véase capítulo "Marcos Conceptuales").

La clave del caos, o más exactamente de la ciencia del caos, es reconocer que las no linealidades son una norma común de la Naturaleza y estudiarlas como tales, no intentando imponer condiciones simplificadoras. La dificultad estriba en que disponemos de muy pocas herramientas matemáticas para poder tratar con no linealidades, hasta tal punto que sólo somos capaces de resolver las ecuaciones no lineales más sencillas y se conocen algunos métodos de resolución, aplicables a pocos casos, consistentes en linealizar las ecuaciones. En consecuencia, al no disponer de herramientas ni de teorías que las pudieran explicar, las no linealidades se han evitado por principio en casi todos los estudios. Fenómenos como las

turbulencias, las tormentas, los láseres, ciertas reacciones químicas, los cambios de estado o la evolución atmosférica se han tratado durante mucho tiempo dejando de lado lo que más tarde ha resultado ser la esencia de su comportamiento: la no linealidad.

Los más entusiastas de la ciencia del caos llegan a afirmar que el siglo XX se recordará por tres cosas: la relatividad, la mecánica cuántica y el caos. Este último, dicen, supone un cambio radical en la concepción de la ciencia y en la forma de ver el mundo.

Históricamente el estudio del caos es relativamente reciente, aunque presente en muchos experimentos científicos conocidos, no se disponía hasta hace poco del lenguaje capaz de expresar lo que mostraba la experiencia. Preguntas como ¿qué es una turbulencia?, ¿por qué existe el orden? o ¿qué sentido tienen las no linealidades?, solían estar fuera del campo de estudio de la ciencia "oficial". No fue hasta la década de los 70 que los científicos empezaron a interesarse por este tipo de fenómenos, y fue aún mucho más tarde cuando se empezó a reconocer el valor de estas teorías.

Como en otros muchos avances de la ciencia, el reconocimiento del caos surge de la aportación de diversas personas desde campos muy diferentes. Mitchell Feigenbaum, físico que desarrolló parte de los pilares básicos de la ciencia del caos, Edward Lorenz, del MIT, primero en observar y destacar el comportamiento impredecible de la atmósfera, Stephen Smale, de la universidad de Berkeley, estudioso de la topología aplicada a sistemas dinámicos y que proporcionó la base matemática para entender ciertas propiedades caóticas, Phillip Marcus, astrónomo en la universidad de Cornell, que reconoció la estructura caótica de la Gran Mancha Roja de Júpiter, Robert May, biólogo que estudió las propiedades no lineales de la ecuación que tratamos en este apartado, Benoit Mandelbrot, matemático de IBM padre de los fractales que han servido como ejemplo y medio gráfico para mostrar el caos, Harry Swinney y Jerry Gollub, que estudiaron la transición de fase entre líquido y vapor y el origen de las turbulencias en fluidos, David Ruelle y Floris Takens, descubridores de los "atractores extraños", fundamentales en el estudio del caos, Michel Hénon, que profundizó en el estudio de los atractores, Joseph Ford y Giulio Casati, que organizaron el primer congreso sobre la ciencia del caos, etc.

Ésta es sin duda una fuente muy importante de complejidad. La aportación que ha hecho la teoría del caos es reconocer la existencia de esta fuente como algo común no como una rara excepción. Además ha contribuido a llegar a algunas consecuencias con amplias repercusiones. Un ejemplo de ello, que ya hemos visto anteriormente, es la predicción del tiempo atmosférico. Durante muchos años se ha estado

investigando para establecer los principios que rigen el comportamiento de la atmósfera y así poder predecirlo. En la base de esta investigación está la idea ya comentada: iguales causas producen iguales efectos. Si se conocen las causas que provocan los diferentes procesos atmosféricos, lluvia, nieve, sol, anticiclones, tormentas, tornados, tifones, huracanes, altas y bajas presiones, etc, bastará con detectar esas causas para concluir qué efectos se van a producir. Disponiendo además de la potencia de cálculo adecuada, hecho que merece una mención especial pues ha conducido, en parte, a los ordenadores paralelos, y del conocimiento necesario, no habría problema para predecir el tiempo.

La ciencia del caos puso fin a esta versión del cuento de la lechera. La atmósfera es un sistema altamente no lineal, donde las condiciones iniciales juegan un papel muy importante pues la más mínima desviación en esas condiciones conduce a efectos totalmente distintos. Esto se conoce como el efecto mariposa y tiene como consecuencia más inmediata que el tiempo no es predecible a largo plazo. Las perturbaciones incontroladas que van apareciendo provocan pequeñas desviaciones del comportamiento previsto, desviaciones que se van acumulando hasta llegar a un punto en que el comportamiento es totalmente diferente. La imposibilidad de conocer todas esas perturbaciones, y mucho menos con precisión infinita, hace imposible que podamos establecer predicciones seguras.

La importancia de esas desviaciones en el comportamiento frente a cambios mínimos en las condiciones iniciales es difícil de entender debido a la "deformación" que nos hace considerar las cosas como esencialmente lineales. El caos no son ligeras variaciones, es la alteración de la propia esencia del comportamiento de un sistema. Hay un ejemplo, clásico ya en la teoría del caos, que ilustra muy bien este hecho:

Una ecuación muy utilizada para modelar la evolución de poblaciones de seres vivos es $Y = r * X * (1 - X)$ donde X es la población inicial e Y la población resultante, r es un parámetro para ponderar esa relación. El término $(1 - X)$ evita que esa relación sea estrictamente creciente e intenta reflejar el hecho de que una población muy elevada tendrá problemas para encontrar alimento o espacio y por tanto no crecerá tanto.

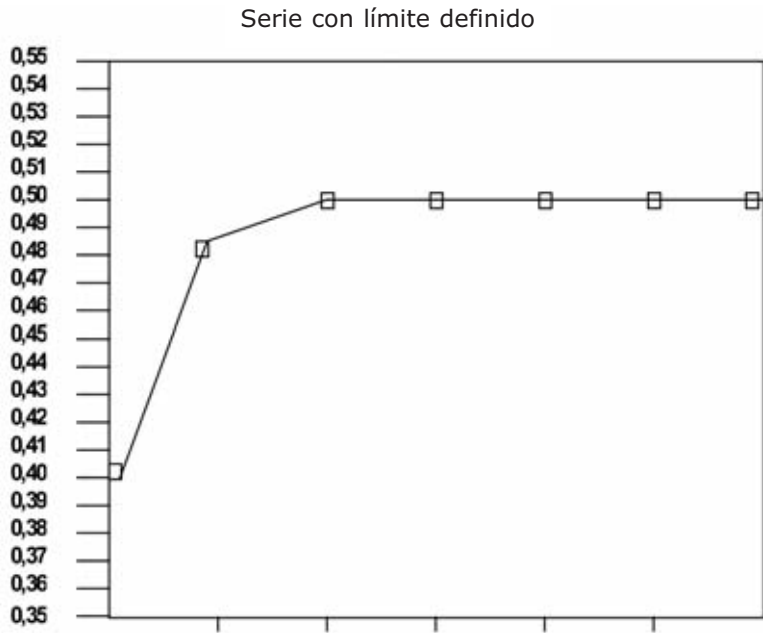


Fig 1. La ecuación tendiendo a un valor final estable

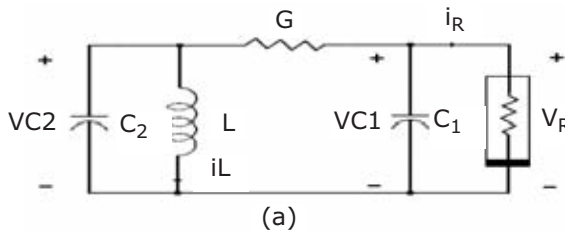
En una primera aproximación se puede suponer que la aplicación iterativa de esa ecuación tenderá a estabilizarse, por ejemplo para $r = 2$ y con una población inicial de 0.4 la serie es (figura 1):

0,4 0,48 0,4992 0,4999987 0,5 0,5 0,5 ...

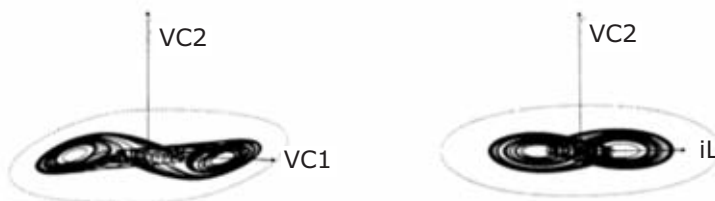
Caos y circuitos electrónicos

Durante mucho tiempo se han asociado las nociones de periodicidad y oscilación por un lado y las de ruido y no determinismo por otro. Esto es lo que se enseña tradicionalmente en las escuelas de ingeniería. Sin embargo, la evidencia ha empezado a demostrar que las oscilaciones no tienen por qué ser siempre periódicas y el ruido puede aparecer en un circuito determinista.

El circuito que aparece en la figura es uno de los más simples que muestran un comportamiento caótico. El único elemento no lineal es la resistencia dependiente de la tensión V_R .



Este circuito, con una resistencia lineal, tendría un comportamiento asintóticamente estable. Sin embargo, la resistencia no lineal actúa como una realimentación de potencia al circuito oscilador (formado por C_2 y L , un circuito tanque típico), provocando el comportamiento caótico. La resistencia limita el crecimiento de la señal dando lugar a trayectorias convergentes (o, como se denomina técnicamente, a trayectorias con atractor). En la siguiente figura se recogen estas trayectorias en forma de proyecciones sobre los planos que se indican. [Matsumoto, 1987].



El comportamiento caótico no es único y exclusivo de circuitos electrónicos o de modelos matemáticos más o menos abstractos. Se ha encontrado este tipo de comportamiento prácticamente en todas las ciencias: astronomía, biología, biofísica, química, ingeniería, geología, matemáticas, medicina, teoría de plasmas e incluso en ciencias sociales. [Chua, 1987]

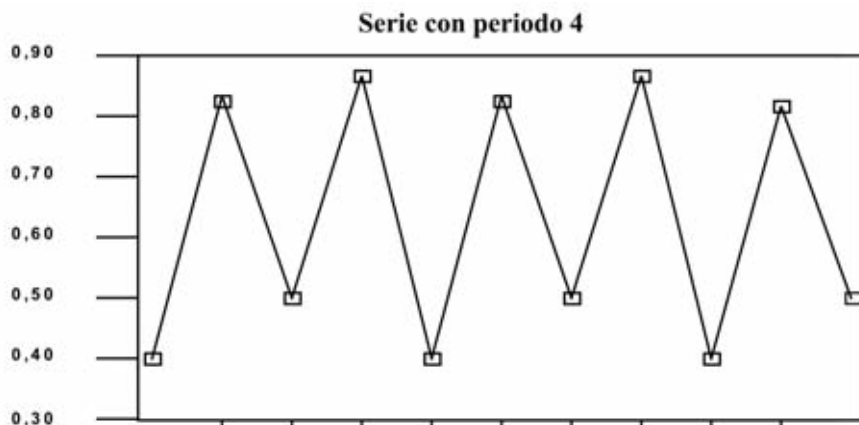


Fig. 2. La ecuación con un comportamiento oscilante de período 4

Y lo mismo sucede para cualquier valor de la población inicial. Este es el comportamiento "normal y esperado" de la ecuación. Pero si se cambia el valor del parámetro r empiezan a suceder cosas muy curiosas, para $r = 3,5$ y con una población inicial de 0,4 la serie es (figura 2):

0,4 0,84, 0,4704 0,8719334 0,3908294 0,8332862 0,4862211
0,8743355 0,3845552 0,8283539 0,497643 ...

que no tiende a ningún valor estable, y no sólo eso, sino que parece oscilar con período 4 -de hecho lo hace-. A esto es a lo que nos referimos con un cambio en la propia naturaleza del comportamiento del sistema, de un sistema estable que tiende a un punto de equilibrio, se ha pasado a un sistema oscilante. Si se varía el parámetro r se llega a otro tipo de sistemas, concretamente se pueden conseguir resultados de período 2, 4, 8, 16, 32 ... y finalmente el caos, una serie de valores que no guardan ninguna relación entre ellos.

Este asombroso comportamiento es aún más sorprendente si se considera que reaparece para diferentes órdenes de magnitud de r . Para representar esta ecuación y su comportamiento se utiliza lo que se denomina un diagrama de bifurcaciones, un tipo de representación que se ha hecho muy común en la teoría del caos. En estos diagramas se representan los valores finales a los que tiende el sistema para distintos valores del parámetro correspondiente. En este caso, para diferentes valores de r se representa un punto si la ecuación se estabiliza, dos si tiene período 2, cuatro si el período se duplica, etc. El resultado es un gráfico como el que muestra la figura 3:

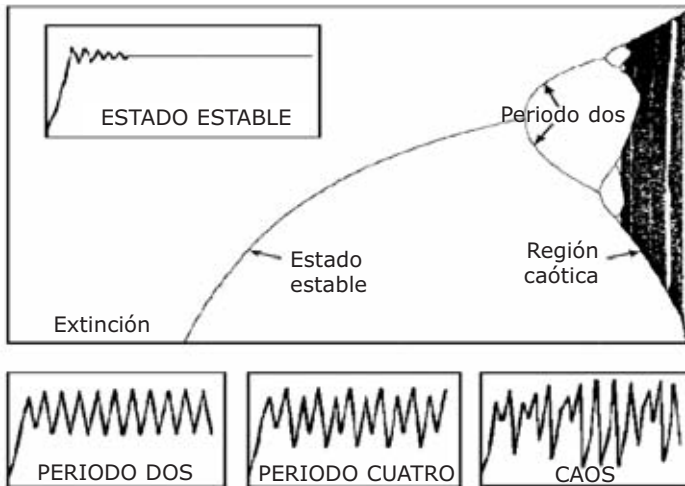


Fig. 3. Diagrama de bifurcación de la ecuación $Y = r * X * (1 - X)$, (Gleick, 1988, p. 79)

En ese diagrama es fácil ver cómo aparece el caos para determinados valores de r . Y también la estructura repetitiva, superado un valor concreto de r el sistema deja de comportarse caóticamente y vuelve a estabilizarse, a medida que r aumenta se repite el proceso apareciendo períodos 2, 4, 8, 32, etc. En la figura 4 se muestra una ampliación de ese diagrama en la que se puede ver cómo se repite la misma estructura dentro de sí misma.

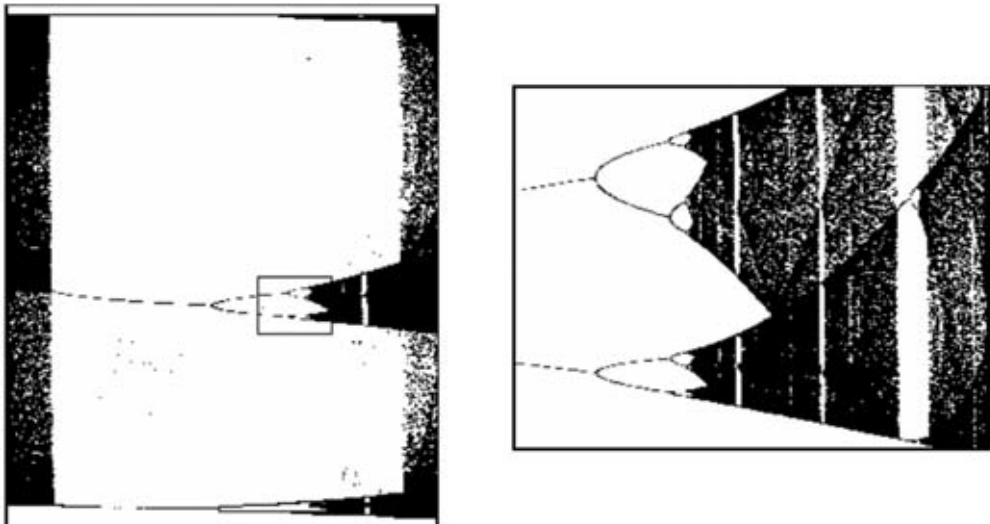


Fig.4. Autoorganización en un sistema caótico. Al forzar el sistema aparece un período 3 que se duplica (6, 12, 24, ...) hasta desembocar en un nuevo caos; la estructura tiene profundidad infinita, [Gleick, 1988, p. 82]

Desde el punto de vista de la complejidad, la consideración del caos es muy relevante pues pone de manifiesto la importancia de las no linealidades y la repercusión que éstas tienen en el comportamiento del sistema. Un ejemplo real se recoge en el cuadro adjunto. También, como en el caso de la atmósfera, indica que hay sistemas que no son predecibles si no se conocen todos los factores que influyen en ellos, hasta el más pequeño, sin que exista la posibilidad de despreciar ninguno ya que hacerlo conduciría a una conclusión errónea. Variaciones mínimas en parámetros que tradicionalmente se han despreciado pueden producir alteraciones radicales en el resultado, como muestra el ejemplo. Esta sensibilidad de los parámetros se convierte así en un factor fundamental en el estudio de ciertos sistemas.

De la estabilidad al crecimiento exponencial

La importancia de las interacciones entre los diferentes factores se da en todas las ramas del saber humano pero en especial en los relacionados con la sociedad. Ésta introduce tal cúmulo de factores y dependencias que dificultan en gran manera las predicciones del comportamiento. Los propios investigadores son conscientes de que la más pequeña alteración produce resultados totalmente diferentes. Ejemplo de ello es un cálculo del crecimiento de la población de los EEUU realizado en 1972. Estableciendo una media de 2 hijos por familia la población tendería a estabilizarse -unos 400 millones hacia el año 2070- pero si se asumía una media de 3 hijos por familia la población no se estabiliza en una cifra mayor -600 o 700 millones, por ejemplo- sino que se produce un crecimiento exponencial con 400 millones en el año 2015 y cerca de 1000 en el 2070. Esta variación en el comportamiento refleja la existencia de un punto crítico en el que el comportamiento se bifurca -estabilidad o comportamiento exponencial- lo que indica una complejidad del sistema mucho mayor de la que se supone a simple vista. Este fenómeno de la bifurcación es típico del estudio del caos y aparece en muchos sistemas naturales. [A partir de "Population and the American Future", N.Y., American Library, 1972, citado en Kemper, 1985].

3. Azar

En su acepción más tradicional, azar es todo aquello que escapa a nuestro control o a nuestra capacidad de predicción. En este sentido es fácil ver el azar como una fuente importante de complejidad y también como una faceta de la complejidad. Está directamente relacionado con el caos pues antes de que existieran los estudios sobre el caos todo comportamiento caótico era considerado como azar y también se puede establecer una relación muy clara entre el azar y la incertidumbre pues como muchos autores han puesto de manifiesto el azar ha sido muchas veces el velo y la excusa que cubría nuestra ignorancia sobre un determinado fenómeno. La controversia que actualmente existe sobre el azar es una de los temas más apasionantes de la ciencia.

Básicamente existen dos posiciones contrapuestas respecto al azar. La primera de ellas es la que considera un Azar, con mayúscula, un Azar "per se", ontológico, inherente a la misma Naturaleza y que por tanto es inevitable. La segunda postura pretende que el azar es sólo el resultado de la falta de conocimiento, es decir, nos resulta azaroso aquello que no conocemos pero no hay azar en las cosas que conocemos. Cada una de ellas refleja una visión diferente del mundo y de la ciencia y se pueden derivar importantes conclusiones. En la discusión sobre este punto entran en juego varias disciplinas como la teoría de probabilidades, la mecánica cuántica o la computación, pues sobre todas ellas influye el hecho de considerar el azar de una u otra forma.

En la mecánica clásica hay una serie de problemas insuperables que ponen de manifiesto, al menos, nuestras limitaciones a la hora de hacer predicciones. Uno de estos ejemplos es el péndulo. Si se sitúa el péndulo en posición vertical, por encima de su punto de suspensión, se desplazará a la izquierda si se le empuja un poco hacia la izquierda o a la derecha si se le empuja un poco a la derecha. Esto es perfectamente lógico y está de acuerdo con la mecánica clásica, pero si fuéramos capaces de colocar el péndulo en esa posición vertical con el mínimo error que nos fuera posible ¿hacia donde caería?, evidentemente unas veces a la derecha y otras a la izquierda con aproximadamente un 50 % de probabilidades de caer a un lado o a otro. La mecánica clásica no puede hacer una predicción mejor que la del 50 %, que no es tal predicción. En estos casos en los que aparece un punto crítico la certeza debe sustituirse por la probabilidad.

Existe además una mecánica estadística en la que el problema de operar con grandes números se soluciona cambiando la pregunta que se hace el investigador: de ¿qué velocidad tiene esta o aquella partícula? se pasa a ¿qué probabilidad hay de encontrar una partícula con una velocidad dentro de un margen especificado?. Se abandona la certeza para evitar la complicación del cálculo. Esto mismo es lo que hacen los meteorólogos cuando dicen que va a llover con un 30 % de probabilidades, no pueden decir más. Sin embargo, luego, o llueve, o no llueve.

La cuestión está en si esas probabilidades reflejan una propiedad intrínseca de la naturaleza, de 100 estados posibles de la atmósfera en 30 llueve, o si esas probabilidades reflejan únicamente una falta de conocimiento, se puede afirmar que lloverá con una certeza del 30 %. Aunque parezca una discusión un tanto sin sentido, elegir una u otra opción conlleva importantes consecuencias. Veamos más ejemplos.

Una de las teorías con más éxito en este siglo es la teoría cuántica. En ella no se maneja certidumbre sino sólo probabilidades. Los estados de un sistema tienen una determinada probabilidad, según la función de onda, pero nunca tenemos una certeza absoluta. El principio de incertidumbre impide que podamos hablar del estado de una partícula ya que postula la imposibilidad de conocer su velocidad y posición al mismo tiempo. Esto es muy importante pues las cosas cuando ocurren son ciertas y completas, o llueve o no llueve. En este sentido sería lógico preguntarse si, existiendo una teoría que proporciona sólo probabilidades, no sería posible encontrar una teoría X que asigne certidumbres. Por ejemplo, la probabilidad de aprobar una oposición es 0,7. Esta sería una afirmación de una teoría probabilística. Pero si estudiamos a cada uno de los que se examinan, su historial y sus aptitudes, somos capaces de decir quién pasará o quién no pasará, lo que corresponde a una afirmación de la teoría X.

Decantarse por una de las dos opciones que presentábamos al principio supone aceptar la existencia de esa teoría X o considerarla imposible. Cuando Einstein dijo aquello de que Dios no juega a los dados estaba intentando subrayar la necesidad de eliminar las probabilidades y sustituirlas por certidumbres, buscando una teoría que fuera más allá de la probabilidad cuántica.

Además, la utilización de una teoría probabilística conlleva ciertas paradojas. Una de ellas es la del gato de Schrödinger. Imaginemos una caja en la que hay un dispositivo radiactivo que se dispara en una determinada dirección y es letal sólo en esa dirección, en cualquier otra es completamente inofensivo. Al meter un gato en la caja hay unas ciertas probabilidades de que el gato esté vivo o de que esté muerto. Al abrir la caja el gato está vivo. La probabilidad se ha hecho certidumbre. La cuestión es ¿cómo es que la probabilidad de estar vivo se ha convertido en certidumbre simplemente por abrir la caja?. La predicción de que el gato estaría vivo puede haber sido del 95 %, es decir, era una buena predicción, pero podría haber sido aún mejor.

A nivel macroscópico el azar también juega un papel importante. Supongamos un dado dentro de un cubilete. Teóricamente es posible establecer qué cara va a salir si somos capaces de calcular todos los movimientos que hará el dado de acuerdo con la forma y la fuerza con que se ha lanzado y todos los parámetros que sean relevantes. En este sentido, el lanzamiento de un dado es determinista y la probabilidad de 1/6 asignada a cada cara sólo nos dice que no podemos en un

momento dado calcular la trayectoria correcta. Pero esta afirmación presenta un problema muy interesante. En principio dice que aunque la computación sea extremadamente costosa es posible calcular qué cara va a salir, siempre y cuando la precisión con que se conocen los datos necesarios sea infinita. Aún en el ordenador más perfecto que se conoce o se pueda concebir existirá un mínimo error, suponiendo que somos capaces de conocer las condiciones iniciales con precisión infinita, y ese error conduce a una incertidumbre sobre la cara que va a salir. Cualquier conocimiento finito conduce a una probabilidad de cada cara de $1/6$.

De nuevo se nos plantea el mismo problema. De poder disponer de un ordenador o de la capacidad de cálculo para ello ¿podríamos establecer qué cara va a salir al lanzar un dado?. Esto implica conocer las condiciones iniciales con exactitud infinita, ¿es posible?. Y, de acuerdo con todo esto, ¿es el lanzamiento de un dado un proceso en el que interviene el Azar o su aleatoriedad es sólo función de nuestra ignorancia?

Desde el punto de vista de la complejidad el azar es un tema muy importante. No cabe ninguna duda de que sí existen casos concretos en que el azar existe sólo en la mente del observador porque no conoce las reglas que rigen un determinado proceso. Es el caso, por ejemplo, de la secuencias pseudoaleatorias que se utilizan en computación. Para cualquier usuario esas secuencias aparecen como aleatorias y sin embargo están generadas por un proceso completamente determinista. Y también es una tendencia muy humana el atribuir al azar todo aquello que no podemos conocer, todo lo que escapa a nuestra ciencia. En este sentido el azar es una complejidad superable a través de una profundización en los mecanismos de comportamiento de una sistema. Pero si se admite la existencia del Azar con mayúscula también se ha de admitir una complejidad inabordable en las cosas. En cualquier caso y al igual que sucede con la complejidad, el azar, con minúscula, depende del observador y no sólo de sus intereses o de la forma de percibir el objeto sino también de su capacidad y de las herramientas de que dispone para hacer frente a la variedad que se percibe en el objeto.

4. Incertidumbre

En lo que atañe a la complejidad la incertidumbre es una de las mayores fuentes con que nos podemos encontrar y también una de las más comunes. En cierta forma engloba al caos y al azar, pues, que se produzcan fenómenos caóticos o azarosos que se traducen en incertidumbre respecto al objeto con que se trata. Pero el caos y el azar tienen un significado concreto mientras que la incertidumbre es algo más bien intuitivo y sin formalizar, al menos en la mayoría de los campos de estudio.

Al tirar un dado se produce una incertidumbre sobre el resultado que saldrá, una turbulencia introduce incertidumbre sobre el comportamiento de un fluido, pero un tipo de incertidumbre que se deja generalmente de lado al no tener herramientas

que permitan manejarla. La mayor formalización sobre la incertidumbre se ha hecho en el campo de la representación del conocimiento y en la inteligencia artificial, donde es necesario reflejar esta realidad de las representaciones que el hombre fabrica de los objetos que le rodean. Así, siguiendo a Klir [Klir, 1988], podemos definir y clasificar la incertidumbre como sigue:

En una primera aproximación incertidumbre tiene un significado amplio que abarca diferentes conceptos resumibles en seis puntos:

1. Conocimiento sin seguridad, cuestionable, problemático
2. Vago, sin definir o determinar
3. Dudoso, falta de algún conocimiento, no seguro
4. Ambiguo
5. No fijo, no constante, con variaciones
6. Con tendencia a cambiar o variar, falta de fiabilidad o seguridad

Estos seis puntos son muy amplios y se pueden relacionar con las definiciones de complejidad que hemos visto, especialmente con la definición de Sáez Vacas, con la que guarda un paralelismo muy curioso, lo que indica la estrecha relación que hay entre la complejidad y la incertidumbre. Discutir cuál es la causa y cuál es el efecto es un círculo vicioso en el que no pretendemos entrar, sólo queremos subrayar que la incertidumbre genera complejidad y la complejidad, incertidumbre.

El conocimiento incompleto de un objeto, la ambigüedad, la falta de fiabilidad o la vaguedad son ideas que están fuertemente ligadas a la noción de complejidad y, al igual que el azar, se pueden relacionar directamente con la complejidad vista como la diferencia de variedades entre observador y el objeto. Y, como el azar, la incertidumbre está asociada al observador nunca al objeto, algo evidente nada más echar un vistazo a los seis puntos anteriores.

Por otro lado, y de acuerdo con Klir [Klir, 1988] esos seis puntos pueden agruparse en dos categorías diferentes: vaguedad y ambigüedad. Por vaguedad se entiende la dificultad de hacer distinciones claras y precisas, la falta de límites concretos. Ambigüedad es más la dificultad de elegir entre varias opciones y está directamente relacionada con las relaciones uno-a-varios.

Cada una de estas categorías lleva asociados varios conceptos similares pero que matizan más su significado concreto. La siguiente lista resume algunos de estos conceptos:

VAGUEDAD	AMBIGÜEDAD
Borrosidad	Falta de especificidad
Falta de claridad	Diversidad
Falta de distinciones	Generalidad
Falta de definición	Divergencia
Falta de limitación	Relaciones uno-a-varios

Algunos de ellos se utilizan profusamente en la literatura técnica, especialmente el de borrosidad. Dentro de la ambigüedad también se pueden distinguir algunas categorías:

1. Ambigüedad en la caracterización de un objeto (falta de especificidad), por ejemplo, decir que alguien es alto no es decir mucho, pues el conjunto de las personas altas es muy amplio y a medida que sea más grande ese conjunto, menos caracterizado estará un elemento que pertenezca a él.
2. Ambigüedad en la pertenencia a una o otra categoría (disonancia en la evidencia), que sucede frecuentemente cuando nos enfrentamos a dos opciones entre las que hay que elegir.
3. Ambigüedad originada por la multitud de opciones (confusión en la evidencia), cuando las opciones son pocas se habla de disonancia, cuando las opciones son muchas introducen su propia ambigüedad al producirse múltiples evidencias al mismo tiempo.

Todos estos matices son muy interesantes cuando se estudia el comportamiento de sistemas pues no es lo mismo enfrentarse con la complejidad que genera un problema mal definido, vago, que la que genera un problema ambiguo, que tiene múltiples soluciones. A la hora de abordar un problema, y por tanto tratar la complejidad que genera, es fundamental identificar cuáles son las fuentes de complejidad que están presentes.

5. Complejidad y complicación

Hay una serie de conceptos que tradicionalmente se asocian con la complejidad hasta el punto de considerarse casi sinónimos. Uno de ellos es la complicación, una característica peculiar de determinados sistemas que debemos distinguir de la complejidad. No cabe duda de que la complicación origina complejidad pero resulta también obvio que no todo lo que es complejo es complicado.

La complicación es un atributo de sistemas artificiales construidos por el hombre que, además, conoce y comprende totalmente su funcionamiento y su estructura [Atlan, 1979]. Lo cual no quiere decir no que tengan errores, simplemente se pretende destacar que son sistemas artificiales en contraposición a sistemas naturales como pueden ser una célula o un organismo vivo. En estos últimos no cabe hablar de complicación sino de complejidad, sin embargo, en una máquina, un ordenador por ejemplo, si se puede hablar de complicación. En este sentido, la complejidad de un algoritmo parte de su grado de complicación. Y es importante no confundirse en este punto. La complicación de un algoritmo se puede medir, por ejemplo, según el número de instrucciones que se necesitan para describirlo en una máquina y con un lenguaje estándar, por ejemplo una máquina de Turing. Según esto, un algoritmo con más instrucciones será más complicado que uno con menos. La complejidad está relacionada con el grado de complicación pero no sólo con él. La complejidad del algoritmo aparece cuando empezamos a considerar parámetros que de alguna manera no dependen directamente del algoritmo, interacción con otros algoritmos, recursos limitados, expresividad del lenguaje de programación, tiempos de computación, características particulares de la máquina donde se ejecuta, optimización del compilador, prevención de errores y fallos, estilo de programación, etc.

La complicación aparece en sistemas en los que conocemos su funcionamiento y su estructura, una diferencia fundamental con la complejidad. El ejemplo del algoritmo es muy claro. De él sabemos lo que hace, clasificar palabras, buscar etiquetas, establecer rutas de comunicación o intercambiar páginas de memoria, y cómo lo hace, su estructura, ventajas y desventajas, supuestos de partida y limitaciones. De acuerdo con ello el algoritmo será más o menos complicado. La complejidad es el resultado de eliminar la abstracción en ese algoritmo y pasar a implementarlo, momento en el cual aparecen todos los condicionantes que antes mencionábamos.

Complejidad algorítmica

La Algorítmica es uno de los campos de estudio más avanzados de la complejidad, tanto es así que en gran parte de las universidades americanas el estudio de la complejidad algorítmica se conoce simplemente por estudio de la complejidad. Dentro de ella se estudian temas tan interesantes como los límites de la computación (cuanta información se puede computar), tiempo de computación de cada algoritmo, tipos de problemas, problemas no resolubles, explosiones combinatorias, etc. Sobre este tema volveremos más adelante cuando tratemos la complejidad del software.

6. Complejidad y desorden

El desorden se puede tratar de muchas maneras, una de las más utilizadas es la entropía. Intuitivamente se suele relacionar el orden con la ausencia de complejidad o al menos con una disminución de ésta. El orden ayuda a entender la estructura y el comportamiento de un sistema y por tanto a hacerlo menos complejo. Pero de ello no se puede deducir que todo desorden implique complejidad. Un gas es un objeto muy desordenado pero ello no implica que sea necesariamente complejo pues podemos tratar ese desorden de forma estadística porque nos conformamos con un cierto conocimiento global, es decir, consideramos la complejidad de una cierta situación. Sobre esta cuestión volveremos en capítulos posteriores. Este hecho ya fue estudiado por Weaver cuando propuso una clasificación de la complejidad en la que consideraba una complejidad desorganizada aquella en la que el grado de desorden y el elevado número de elementos permitían un tratamiento estadístico (ver en el capítulo Marcos Conceptuales).

Un sistema complejo presenta un desorden que podríamos llamar virtual, un desorden aparente que oculta un orden subyacente. El movimiento de los planetas se consideraba desordenado, o aleatorio, mientras no se estableció una teoría que lo explicase. La complejidad puede verse entonces como un orden del que no conocemos la clave [Atlan, 1979]. Esto nos permite enlazar con los conceptos que antes veíamos. El desorden está íntimamente ligado a la noción de azar, el azar produce desorden y lo desordenado es aleatorio. Podemos hacernos entonces la misma pregunta que nos hacíamos antes, ¿es el desorden una característica propia de la Naturaleza o sólo un nombre con el que subrayamos nuestra incapacidad para encontrar el orden que hay en todas las cosas?.

Siguiendo la propuesta de Atlan, la complejidad como un orden desconocido, entramos de lleno en el terreno de la incertidumbre que podría interpretarse como una medida del desconocimiento que tenemos de un determinado orden. Lo que no conocemos aparece como desordenado, aleatorio y aumenta nuestra

incertidumbre sobre su comportamiento. El orden, por el contrario, implica una pauta, unas normas, una estructura que rige en todo momento y que permite reducir esa incertidumbre, abandonar el caprichoso azar y reducir por tanto la complejidad.

7. Complejidad y redundancia

Este es el último de los conceptos que vamos a estudiar aquí y nos va a servir fundamentalmente para relacionar los anteriores entre sí. La redundancia consiste en que el conocimiento de un elemento nos aporta una cierta cantidad de información sobre los demás elementos del sistema. Este hecho nos lleva a una conclusión muy interesante: si hay redundancia percibimos un cierto orden. La repetición de una determinada pauta permite evitar el desorden del que antes hablábamos y por tanto reduce la incertidumbre, la aleatoriedad y finalmente la complejidad.

La redundancia puede verse entonces como una medida de la simplicidad y el orden. Y en principio podría considerarse que la redundancia reduce la complejidad si no fuera porque implica un aumento de la complicación cerrándose así un círculo que nos relaciona estos tres últimos conceptos. La redundancia permite establecer un cierto orden y reducir la incertidumbre pero conlleva un aumento considerable de la complicación. Será, pues, un problema de compromiso elegir el grado de redundancia que nos permite establecer una pauta pero no aumenta en demasía la complicación del sistema.

Hay una serie de ejemplos muy interesantes que ilustran un serie de características de la redundancia que nos pueden servir muy bien para cerrar la discusión sobre todos estos conceptos. El primero de ellos son los arabescos, una decoración en la que unos pocos patrones sencillos se repiten una y otra vez formando estructuras muy complicadas. En este caso la redundancia no está tan relacionada con la información pero sí con una repetición física de un motivo único un gran número de veces. La complicación en el resultado y el desconocimiento de las pautas seguidas por el artista para establecer las reglas según las cuales se repiten los motivos dan lugar a la complejidad que percibe el observador. Sin entrar en mayores profundidades, lo que dijimos de la complicación también vale aquí, la redundancia en sí no implica complejidad, es la interacción con elementos ajenos la que genera esa complejidad. En el caso de los arabescos hay una serie de teorías que sostienen que los patrones seguidos se basan en conceptos abstractos y espirituales que dictan las normas y las pautas a seguir y que, en lo que a nosotros nos interesa, son los que introducen la complejidad en esos arabescos.

Otro ejemplo son los fractales, estructuras redundantes por excelencia, y que cualquiera que haya tenido la oportunidad de contemplar imágenes de fractales como el conjunto de Mandelbrot puede atestiguar la complicación que hay en ellos. Complicación que se transforma en complejidad cuando no conocemos el algoritmo, generalmente muy sencillo, que ha servido para trazarlo.

8. Resumen

Podíamos extendernos mucho sobre cualquiera de los seis conceptos que hemos tratado en este capítulo. Algunos de ellos, como el caos o el azar, son temas de candente actualidad y sobre los que se han escrito libros enteros. Evidentemente aquí no podemos ni pretendemos hacer lo mismo puesto que nuestro principal objetivo es el estudio de la complejidad.

En función de ésta, azar, caos, incertidumbre, redundancia, complicación y desorden aparecen como su origen. Es decir, se pueden ver como fuentes de complejidad que podemos añadir a las que habíamos considerado anteriormente. Pero lo más destacado de ellas es que están profundamente interrelacionadas y es difícil hablar de una de ellas sin mencionar a las demás. La incertidumbre y la complicación son las que más directamente se relacionan con la complejidad y en este sentido conviene recordar el paralelismo que existe entre la definición de incertidumbre de Klir y la definición de complejidad de Sáez Vacas. Por otro lado, el azar es quizá la idea más cercana a cualquiera y también la más intuitiva pero podemos subordinarla de alguna forma a la incertidumbre.

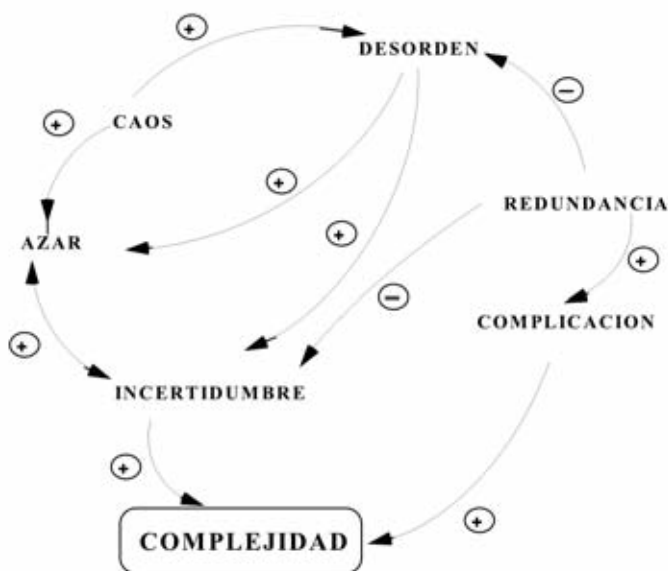


Fig. 6. Cuadro de relaciones entre los seis conceptos estudiados en este capítulo

Como resumen de todo esto podemos pensar en un diagrama como el que muestra la figura 6 en el que hemos intentado reflejar las relaciones entre los conceptos estudiados. Las flechas llevan un signo + ó - según que la presencia de una de estas características conduzca a aumentar o disminuir la presencia de otra. Por ejemplo, la aparición de fenómenos caóticos implica un aumento de desorden, la existencia de redundancia implica una disminución del desorden y un aumento de la complicación, etc.

Queda aún mucho campo de estudio en lo referente a estos temas. El caos se produce generalmente de forma redundante, es decir, el mismo fenómeno se produce a diferentes escalas -recordar el ejemplo visto-, lo que introduce un factor de redundancia que sin duda ayuda a disminuir aún más el desorden. El paralelismo que existe entre la incertidumbre y la complejidad también es muy interesante desde el punto de vista conceptual y desde el punto de vista práctico algunas de las formalizaciones hechas para la incertidumbre se podrían intentar aplicar a la complejidad.

9. Bibliografía

Dividida en dos apartados. En el apartado de Notas Bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

Sobre el Caos: existe una bibliografía amplia, tanto en forma de libros como de artículos, dado que se trata de un tema de moda. En español y como obra introductoria para los que no deseen perderse en complicados formalismos matemáticos está **Caos**, de James Gleick, algo falto de rigor al estar escrito por un periodista, pero por eso mismo sencillo y ameno de leer, [Seix Barral, Barcelona, 1987]. Otro título interesante, sobre todo para aquéllos con una formación más ingenieril o científica, también en español, es **Fórmulas del éxito en la Naturaleza**, de Hermann Haken [Biblioteca Científica Salvat, Barcelona, 1986]. Existe un número de los **Proceedings del IEEE** dedicado a sistemas caóticos y es muy interesante para profundizar en el aparato matemático y teórico de las manifestaciones del caos [Proceedings of the IEEE, Agosto 1987].

Sobre el Azar: las referencias básicas que hemos utilizado son **Proceso al Azar e Ideas sobre la complejidad del mundo**, ambos de Jorge Wagensberg [en la colección Superínfimos de Tusquets Editores], en esta misma colección se pueden encontrar títulos muy sugerentes que no se limitan a este tema y muy relacionados con el estudio de la complejidad. "Proceso al Azar" es el resumen de una serie de conferencias pronunciadas durante un encuentro en Figueras, que reunió a

importantes personajes de la ciencia y el pensamiento en torno al problema del azar, puede ser interesante para los iniciados en la física y las matemáticas y para los interesados en la filosofía especulativa. "Ideas sobre la complejidad del mundo" tiene un objetivo más amplio y trata muchas ideas relacionadas con la complejidad, entre ellas el azar, es relativamente sencillo de leer y con ideas muy estimulantes.

Sobre Incertidumbre: existe también una bibliografía muy amplia, aunque no directamente relacionada con los aspectos que aquí hemos tratado. La incertidumbre suele tratarse en profundidad en la mayoría de los libros sobre Inteligencia Artificial y principalmente en los que se ocupan de la lógica borrosa. El que hemos tomado como referencia **Fuzzy sets, Uncertainty and Information**, de George Klir (al que hemos citado con anterioridad por sus trabajos sobre sistemas y complejidad) resulta algo engorroso pues está orientado al estudio formal y matemático de la lógica difusa, el capítulo 5 es lo más interesante desde el punto de vista que aquí nos ocupa. Existe un artículo de Klir que puede resultar interesante y más sencillo de leer: "The role of Uncertainty Principles in Inductive Systems Modelling" [Kybernetes, 17, 2], gran parte de lo que aparece en este artículo constituye el capítulo 5 del libro anteriormente citado.

El resto de los conceptos que hemos tratado se encuentra muy disperso en la bibliografía, por ello nos limitaremos a dar sólo algunas ideas sobre referencias que tocan estos temas, aunque ninguna lo hace en exclusiva. La utilizada como referencia directa en este capítulo es **Entre le cristal et la fumée**, de Henri Atlan, en francés (hay traducción al español), un libro donde aparecen muchas ideas sobre sistémica y complejidad. Otro libro interesante pero difícil (existe versión en español), premio Pulitzer, es **Gödel, Escher, Bach: An eternal golden braid**, de Douglas, R. Hofstadter [Basic Books, Nueva York, 1979]. Otro libro interesante como ejemplo de la complejidad conseguida a través de formas simples puede ser **Las formas del color**, de Karl Gerstner [Hermann Blume, Madrid, 1988].

Referencias bibliográficas

Atlan, H. (1979) **Entre le cristal et la fumée, essai sur l'organisation du vivant**, Editions du Seuil, Paris, (hay traducción al español).

Gerstner, K. 1988 **Las formas del color**, Hermann Blume, Madrid.

Gleick, J. (1987) **Caos**, Seix Barral, Barcelona.

Haken, H. (1986) **Fórmulas del éxito en la naturaleza**, Biblioteca Científica Salvat, Barcelona.

Hofstadter, D.R. (1979) **Gödel, Escher, Bach: An Eternal Golden Braid**, Basic Books, Nueva York, (hay traducción al español).

Klir, G.J.(1988) **Fuzzy sets, Uncertainty and Information**, Prentice-Hall International, 1988.

Klir, G.J. "The role of uncertainty principles in Inductive Systems Modelling", **Kybernetes**, 17, 2, pp. 24-34.

Proceedings of the IEEE (1987), Agosto.

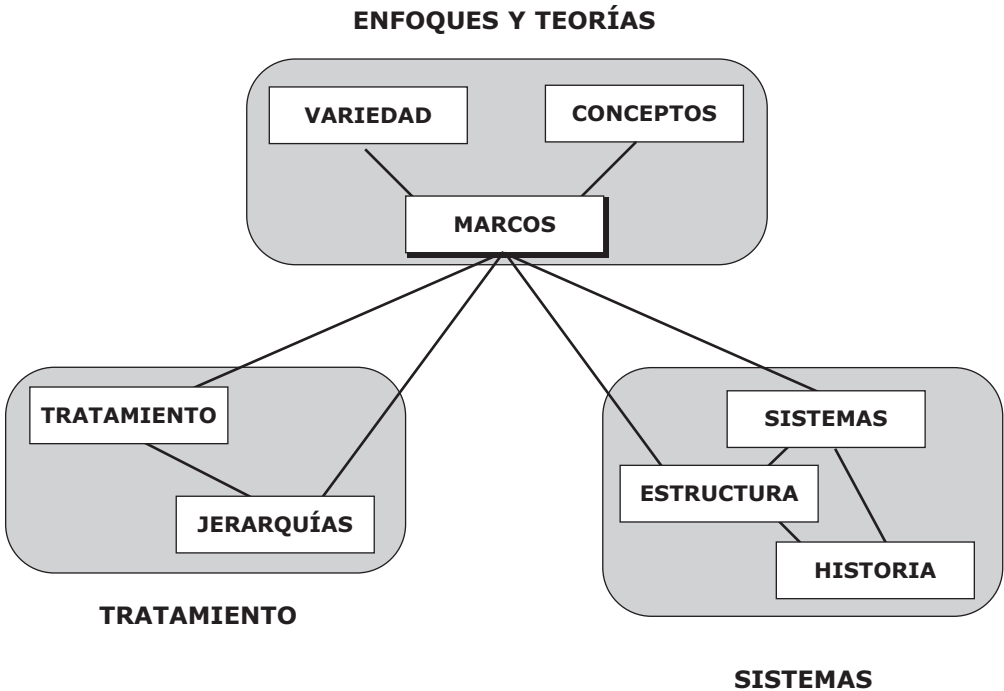
Wagensberg, J. (1986) **Proceso al Azar**, Colección Superínfimos de Tusquets Editores, Barcelona.

Wagensberg, J. (1985) **Ideas sobre la complejidad del mundo**, Colección Superínfimos de Tusquets Editores, Barcelona.

Capítulo 3: Marcos conceptuales

¿Qué es exactamente lo que se entiende por complejidad?. ¿Cómo podemos formalizar la idea intuitiva que tenemos de este término?. Como veremos, diversos autores han realizado un esfuerzo por crear un marco donde definir la complejidad. En este capítulo realizaremos una aproximación a unos cuantos de estos esfuerzos.

Se estudiarán, entre otros asuntos, algunas clasificaciones de la complejidad (los tres rangos de Weaver, la T de Flood, el modelo de tres niveles de Sáez Vacas,...), los atributos que la caracterizan (importancia del observador, consideración del objeto como sistema,...), y una clasificación de sistemas (Klir) que da lugar a diferentes formas de entender la complejidad.



1. Introducción

"...[el término] **complejidad** tiene varios significados posibles..." [Klir, 1985].

"En general, parece que asociamos **complejidad** con algo que nos parece difícil de entender" [Flood, 1987].

"La **complejidad** de un objeto está en los ojos del observador" [Klir, 1985].

"La **complejidad** es un concepto con el que nos enfrentamos en todas las actividades humanas" [Sáez Vacas, 1990].

"...un concepto que la ciencia moderna intenta aprehender sin haber conseguido todavía definir satisfactoriamente. Es la **complejidad**" [Wagensberg, 1985].

"...no hay todavía una prueba convincente de que las tareas realizadas por los modernos ordenadores [...] no podrían ser realizadas igual de bien [...] con la mitad de componentes. Esta posibilidad ha dado lugar a una nueva disciplina matemática llamada teoría de la **complejidad**" [Pippenger, 1978].

Observemos un momento estas citas, tomadas de diversos autores que se han ocupado de trabajar sobre la complejidad. Unas hablan de la importancia de la complejidad. Otras realizan alguna observación sobre su significado. Pero después de analizarlas con un poco de detalle, nos surgirá la duda de qué es exactamente lo que se entiende por complejidad en cada uno de los casos. A medida que vayamos profundizando en el tema, podremos observar que hay varias formas de entender este concepto. En este capítulo realizaremos una aproximación a unos cuantos esfuerzos destinados a obtener un marco global donde encajar y definir el término "complejidad", y otros con él relacionados.

Comenzaremos por un enfoque, debido a Weaver, donde los sistemas se clasifican según ciertas características estructurales. A partir de esta clasificación, obtendremos una primera idea de las características que tiene lo complejo.

Según Klir, la complejidad está íntimamente ligada a los sistemas, y por tanto para poder estudiarla adecuadamente, es necesario primero realizar una clasificación y definición adecuada de éstos. Así, estudiaremos la complejidad de los objetos a través de su clasificación jerárquica como sistemas.

Para Flood la complejidad está ligada tanto al objeto observado como al observador (idea, por otra parte, ya adelantada por Klir). A partir de esta premisa, estudia las características de cada uno de estos dos elementos, y cómo influyen en la aparición de complejidad. Así llegamos a un "desensamblado por niveles" de la complejidad. Además, realiza una clasificación de las ciencias que en cierta forma extiende la división de los sistemas de Weaver, al considerar el elemento humano.

Sáez Vacas, además de aportar una definición extensiva, propone una diferenciación de la complejidad en tres niveles. Este modelo en niveles es aplicable de forma muy directa, entre otros, al campo de las tecnologías de la información.

2. Warren Weaver. Una clasificación de la complejidad

Una de las clasificaciones más clásicas de la complejidad es la propuesta por Warren Weaver [Weaver, 1948], basada en el tipo de tratamiento matemático que requieren los sistemas para su estudio y tomando como punto de partida dos de sus características estructurales:

- a. Por una parte, el número de elementos (o variables) que componen el sistema. Estableciendo un rango que va desde los que están formados por muy pocos elementos (como pueden ser los modelos del sistema solar, los problemas de cinética clásica, etc.) hasta los que tienen multitud de ellos (como los gases constituidos por moléculas, los ordenadores basados en millones de transistores, o el modelo de ser vivo visto como un conjunto de células).

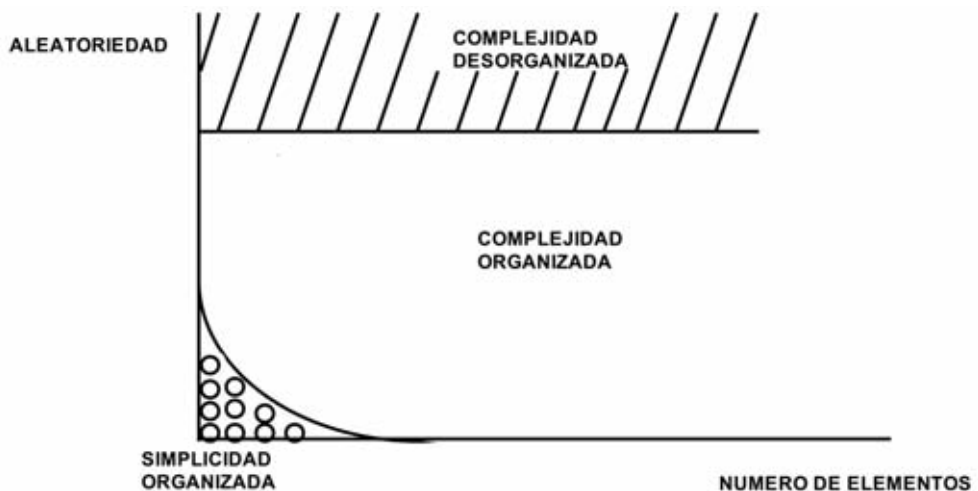


Fig.1 Los tres rangos de complejidad propuestos por Weaver

- b. Por otra, el grado de aleatoriedad observado en el comportamiento de los elementos del sistema. Aquí nos moveremos desde el determinismo puro y duro de los sistemas físicos newtonianos hasta la imprevisibilidad total (a nivel microscópico) de los modelos de la mecánica estadística aplicada (por ejemplo) a los gases.

Los tres rangos de complejidad propuestos por Weaver son los siguientes:

- a. Simplicidad organizada.** Weaver incluye aquí los sistemas que están compuestos por un número pequeño de variables, dependientes unas de otras de una forma fuertemente determinista. Este es el tipo de sistemas estudiados normalmente por la ciencia hasta finales del siglo XIX. La forma en que avanzaba el conocimiento científico en esa época tenía que ver sobre todo con el descubrimiento de un modelo simple que era utilizable para explicar fenómenos a primera vista mucho más complejos. Esta simplificación normalmente se conseguía despreciando muchas variables, suponiendo que su contribución al efecto observado era mínima. Así se conseguía también diseñar experimentos minimizando la influencia de esas variables despreciadas, para corroborar las teorías. Este tipo de sistemas es perfectamente adecuado para su tratamiento con métodos del análisis matemático (ecuaciones diferenciales, etc.).
- b. Complejidad desorganizada.** Este rango se da en sistemas donde las variables se relacionan de una forma muy aleatoria. Comenzaron a estudiarse a finales del siglo XIX, con los trabajos sobre las moléculas de un gas en un espacio cerrado. El número de variables implicadas es sencillamente inimaginable: del orden de 10^{23} moléculas. Además, se mueven muy rápidamente, y el gran número de colisiones que sufren hace que su comportamiento individual sea en la práctica impredecible. Muy pronto se comprendió que estos sistemas no eran abordables con los métodos desarrollados hasta entonces (usados con éxito en los problemas de simplicidad organizada). La solución vino de la mano de los métodos estadísticos, debidos entre otros a los trabajos de Gibbs. Se pasa de considerar las características de cada partícula a estimar unos valores medios para ciertas variables de sistema. Este enfoque se mostró tanto más apropiado cuanto mayor era la aleatoriedad de los elementos del sistema considerado.
- c. Complejidad organizada.** Así pues tenemos métodos para tratar con los dos extremos del diagrama de Weaver. Pero ¿qué pasa con la amplia zona intermedia?. Aquí tenemos un gran número de sistemas cuyo estudio a menudo interesa a varias disciplinas, como son los que tratan las ciencias de la vida, de la sociedad, del medio ambiente... Todos ellos comprenden un gran número de aspectos que no pueden ser despreciados (al menos sin desvirtuar fuertemente la validez del modelo), con lo que se salen del alcance del análisis matemático usado en los problemas de simplicidad organizada. Pero tampoco tienen el grado de aleatoriedad suficiente como para que los métodos estadísticos sean de aplicación. Así pues, se hace necesario descubrir y aplicar nuevas formas de simplificación para tratar con estos problemas. Este es el tipo de complejidad al que nos referiremos a partir de ahora.

Consideremos un programa de ordenador medianamente grande. Supongamos que está en un lenguaje X, y que está compuesto por N instrucciones distintas. Cualquiera que haya programado alguna vez sabe que si ese N aumenta, la dificultad para entender el programa aumenta también, pero mucho más rápido. Si el lenguaje no está diseñado para evitarlos, aparecen los efectos colaterales: si cambiamos un pequeño grupo de instrucciones en una parte del código, no sabemos con exactitud qué efecto tendrá este cambio sobre otras partes del programa. Los lenguajes de programación intentan evitar esto mediante la definición de procedimientos, funciones, módulos, etc. Desde los lenguajes cercanos a la máquina (ensambladores) hasta los más modulares (ADA, MODULA-2) hay un largo camino intentando jerarquizar al máximo posible los programas. Puede entenderse este esfuerzo como una tentativa de sacar los programas del terreno de la complejidad organizada, en el que se encuentran "naturalmente" por tener un alto número de componentes (instrucciones) relacionadas determinísticamente (en el ordenador cabe muy poca aleatoriedad), llevándolos hacia la simplicidad organizada (pocos módulos "cerrados", relacionados de forma sencilla).

3. George J. Klir. Complejidad y Jerarquías de sistemas

Es importante señalar que Klir es un destacado "sistemista" y son muy conocidos sus trabajos en el campo de la formalización matemática de los mismos. Aquí no vamos a entrar en este apartado sino que nos vamos a centrar en la complejidad teniendo en cuenta siempre el sesgo que tienen las propuestas de Klir. Como se recoge en los apartados en que se va a subdividir este breve estudio, la complejidad de la que habla Klir es la complejidad de los sistemas que casi utiliza como una excusa para introducir el tema más importante de las jerarquías epistemológicas.

3.1. Complejidad en los objetos

El término "complejidad" es difícil de manejar porque tiene muchos significados posibles. Reconociendo esta realidad, Klir se plantea estudiar los aspectos invariantes de la complejidad cuando recorremos las muchas caras que presenta. Como primer paso propone una definición de complejidad extraída directamente del diccionario:

"Complejidad: la cualidad o estado de algo complejo, i.e.,

- con muchas partes o elementos o estructuras interrelacionadas y, en consecuencia, difícil de entender, o

- caracterizado por muchas partes, aspectos, detalles o nociones y exigiendo un profundo estudio o examen para ser entendido o tratado."

Esta definición es en principio aplicable a todas las cosas, materiales o abstractas. Y las dos primeras consecuencias que podemos extraer de ella son las siguientes: primero, que el grado de complejidad parece asociarse con el número de partes y de interrelaciones reconocibles y segundo, que la complejidad tiene una connotación subjetiva dado que se hace depender de la habilidad del observador para entender o manejar el objeto en consideración.

La complejidad depende de la interacción entre una persona y el objeto y por tanto de los intereses y la forma de interactuar con el objeto que escoja la persona. La complejidad está en los ojos del observador. Dado que las formas de interactuar con un objeto son virtualmente infinitas, hay por tanto infinitas formas en las que puede manifestarse la complejidad. Esta está en función de la imagen que se forme del objeto a partir de una serie de atributos observables en él. Según Klir, estos atributos no son directamente accesibles al observador sino sólo a través de sus imágenes abstractas, resultado de la percepción y de un tipo de medida específico. A estos atributos Klir los llama variables, y cuando se dispone de un conjunto de variables referidas a un mismo objeto se dice que se ha establecido un sistema. Un sistema es una forma de ver el mundo.

Esto tiene una consecuencia muy importante, dado que no se está tratando con los objetos sino con imágenes de ellos (sistemas), y por tanto no es operativamente significativo ver la complejidad como una propiedad intrínseca de los objetos. La noción relevante es la de complejidad en los sistemas.

De aquí que Klir se plantee como una tarea primordial estudiar cómo se desarrollan las imágenes del mundo con las que trabajamos. Es decir, cuáles son los procesos que conducen al desarrollo de un sistema y qué abstracciones los caracterizan. La respuesta de Klir es las jerarquías epistemológicas de sistemas pero también adelanta que no es una solución única pues depende del marco conceptual en el que estemos trabajando. Para diferentes sistemas dentro del mismo marco conceptual se deben aplicar diferentes significados de la complejidad.

3.2. Jerarquía epistemológica de sistemas

Klir empezó a desarrollar este trabajo en la década de los 60, partiendo de la noción de sistema en diversas disciplinas. Esta jerarquía se utiliza hoy en día en un sistema experto diseñado para tratar problemas de sistemas.

Dicho de una forma general, la formulación de un problema en una disciplina científica cualquiera es en realidad la adopción, por un investigador, de un punto de vista sobre ese problema [Orchard, 1972]. El científico definirá y aislará el problema, con

lo que "creará" el objeto de estudio, que ya es una modelización de la realidad. Sobre este objeto realizará experimentos, recogerá datos, hará análisis teóricos... Intentará encontrar un modelo que permita explicar su comportamiento, estableciendo una correspondencia entre objeto y modelo. El paso de unos modelos a otros, cada vez más generales, nos hace ir definiendo sucesivos sistemas a partir del objeto estudiado, que no son sino estados o etapas para una vía de conocimiento del objeto. Cada uno de estos sistemas pertenecerá a lo que Klir denomina un "nivel epistemológico" diferente.

La diferencia entre un sistema de un nivel epistemológico y otro de un nivel superior es el nivel de conocimiento asociado a las variables que lo definen. Estas variables se establecen en el nivel inferior. Un sistema en un determinado nivel comprende el conocimiento de todos los sistemas de nivel inferior y, al mismo tiempo, posee un determinado conocimiento que no está en los niveles inferiores.

En el nivel epistemológicamente más bajo, denominado nivel 0, un sistema viene definido por un conjunto de variables, un conjunto de estados posibles (valores) conocido para cada variable y algún método operacional para describir el significado de las variables y sus estados en términos de los atributos asociados del mundo real y sus manifestaciones.

El conjunto de variables se divide en dos, el conjunto básico y el conjunto de apoyo. Ejemplos de variables de apoyo son las que representan el tiempo, el espacio o poblaciones de individuos de la misma categoría. Estos sistemas se denominan también Sistemas Fuente por su capacidad potencial de generar datos.

Cuando al sistema fuente se le añaden una serie de datos, esto es, una serie de estados concretos de las variables básicas dentro del conjunto de apoyo, se pasa a considerar un sistema de nivel epistemológico 1. Este tipo de sistemas se denomina también sistema de datos. Por lo general, los conjuntos de datos que se asocian al sistema fuente para formar el Sistema de Datos se pueden representar de acuerdo con alguna función, por ejemplo, una función del tiempo.

Los niveles superiores se caracterizan por algún aspecto invariante de los límites impuestos a las variables básicas. Estos límites se pueden utilizar para generar estados concretos de las variables.

Así, el nivel epistemológico 2 es el de los sistemas representados por alguna característica invariante (invariante con el tiempo, invariante en el espacio, invariante con la población ...) de los límites impuestos a las variables básicas del sistema fuente y, posiblemente, por algunas variables adicionales. Cada una de estas variables adicionales se obtiene a partir de una variable básica y una transformación en las variables de apoyo. Este tipo de sistemas se denominan también Sistemas Generativos ya que los límites impuestos pueden utilizarse para describir procesos en los que se generan estados de las variables básicas.

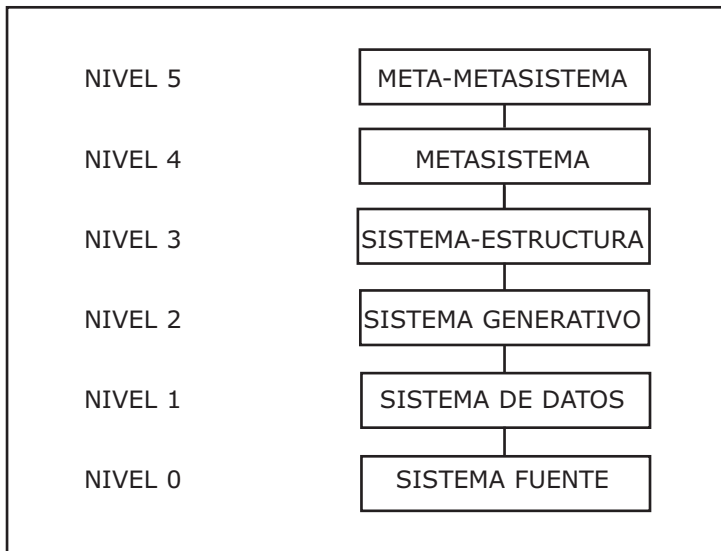


Fig. 2. Jerarquía de sistemas, según Klir

Ejemplo de este tipo de sistemas son las ecuaciones diferenciales con coeficientes constantes, las cadenas de Markov o las máquinas de estados finitos. Cada uno de ellos caracteriza una limitación de las variables, un límite invariante. La solución, por ejemplo, de una ecuación diferencial con condiciones de contorno específicas representa un conjunto de datos.

El tercer nivel epistemológico está formado por todos aquellos sistemas definidos en función de un conjunto de sistemas generativos vistos como subsistemas de un sistema genérico más amplio. Estos subsistemas pueden estar acoplados (compartir variables) o interactuar de alguna manera. Este tipo de sistemas se denomina Sistema-Estructura y se representa típicamente por diagramas de bloques, siendo cada uno de los bloques un sistema generativo.

El cuarto nivel epistemológico, y siguientes, se caracterizan porque los subsistemas inferiores pueden cambiar de acuerdo con alguna de las variables del conjunto de apoyo del sistema fuente. Según el tipo de cambio permitido se tratará de un sistema de nivel 4, metasistema, de nivel 5, meta- metasistema, de nivel 6, meta-meta-metasistema, etc.

Ejemplo de metasistemas son máquinas de estados finitos en los que se permite una variación con el tiempo. La máquina de estado finito en sí es un sistema-estructura. Cuando se permite que ese sistema cambie de acuerdo con alguna norma establecida se pasa a considerar un metasistema, que se puede ver como una colección de sistemas cada uno correspondiente a un valor concreto de la

variable que controla el cambio, por ejemplo el tiempo. Así, un metasistema se puede ver como un conjunto de sistemas-estructura, uno para el momento 1, otro para el momento 2, etc.

Los metasistemas, de cualquier orden, son importantes pues permiten estudiar fenómenos cambiantes como pueden ser la adaptación, autoorganización, morfogénesis, autopoiesis, evolución, etc.

Una forma más de representar esta jerarquía y todas las posibles combinaciones de sistemas que se pueden conseguir es la que se muestra en la figura 3. Los tipos individuales de sistemas se representan por F, D, G, y M (Fuente, Datos, Generativo y Metasistema), mientras que los representados por dos o más caracteres representan combinaciones de los tipos individuales. Por ejemplo, ED es un sistema-estructura compuesto de sistemas de datos, MEG es un metasistema compuesto de sistemas-estructura que a su vez se descomponen en sistemas generativos, etc.

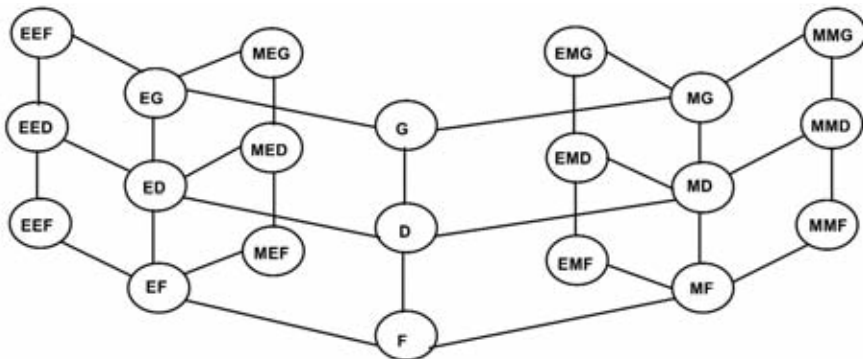


Fig. 3. Algunos de los tipos básicos de sistemas epistemológicos (Klir, 1985)

Para terminar con las jerarquías epistemológicas, vamos a ver un ejemplo de aplicación de estos conceptos a un sistema concreto: una resistencia.

Sistema Fuente: En el que debemos distinguir unas variables relevantes, unos estados y un procedimiento para relacionar las variables con el circuito real:

VARIABLES: la tensión en bornas (V), la corriente que atraviesa el componente (I) y la resistencia eléctrica (R). Como variable de soporte tomamos el tiempo (t).

ESTADOS: La tensión y la corriente pueden ser cualquier número real, el resto sólo números reales positivos.

PROCEDIMIENTO: para ligar estas variables con el circuito real establecemos un procedimiento de medida con un polímetro para la tensión y la corriente, y los datos aportados por el fabricante para la resistencia.

Sistema de Datos: Consiste en establecer un conjunto de datos válidos para esas variables y esos estados. En nuestro caso, mediremos en diferentes momentos (diferentes valores de t) la tensión y la corriente (al aplicarle diferentes excitaciones en bornas). Estas medidas podrían representarse por medio de tablas (si las medidas se toman en instantes discretos) o de gráficas en función del tiempo (si se mide de forma continua). Los valores obtenidos constituyen el sistema de datos.

Sistema Generativo: Para el que elegimos una ecuación que representa el comportamiento genérico de una resistencia (ley de Ohm). Es un sistema generativo porque a partir de esta expresión podemos obtener los diferentes comportamientos (estados de las variables) del sistema para situaciones particulares (por ejemplo, otras excitaciones). El sistema de datos anterior puede obtenerse con gran aproximación resolviendo esta ecuación para valores concretos de tensiones, corrientes y resistencias.

Sistema-Estructura: Estará compuesto por varios sistemas generativos del tipo del hasta aquí analizado. Al interconectar de una determinada forma nuestra resistencia con condensadores, transistores, fuentes, etc., formando un circuito electrónico, obtendremos un sistema-estructura.

Metasistema: Son aquellos sistemas en los que la variación de una determinada variable determina diversos sistemas-estructura. En nuestro caso (circuito electrónico) hay un metasistema determinado por la variable tiempo. Para t pequeño, tenemos un estudio en régimen transitorio, mientras que para t grande, el circuito se considera en régimen permanente. Estos dos casos dan lugar a una definición diferente del sistema-estructura. Así, los circuitos equivalentes y las simplificaciones que se pueden asumir en uno y otro régimen son diferentes, lo que hace que los sistemas-estructura también lo sean.

3.3. Principios generales de la complejidad de sistemas

Una vez vistos los diferentes tipos de sistemas que se pueden establecer, el siguiente paso es identificar la complejidad asociable a cada uno de ellos. Como los sistemas tienen muchas caras, la complejidad asociada a ellos tendrá también muchas caras. Pero dentro de esta diversidad se pueden establecer dos principios generales de la complejidad de los sistemas:

De acuerdo con el primer principio, la complejidad de un sistema de cualquier tipo debe ser proporcional a la cantidad de información necesaria para describirlo. Información en su sentido sintáctico, no semántico o práctico. Una forma de medir

esta complejidad es considerar el número de entidades consideradas en el sistema (variables, estados, componentes) y las interdependencias entre ellos.

De acuerdo con el segundo principio, la complejidad de un sistema debe ser proporcional a la cantidad de información necesaria para resolver cualquier incertidumbre asociada con él. Información, de nuevo, en su sentido sintáctico. Las medidas de incertidumbre están muy desarrolladas (ver el capítulo correspondiente) y permiten establecer este tipo de complejidad de una forma muy clara.

4. R. Flood. Una nueva dimensión de la complejidad

El punto de partida elegido por Flood es la siguiente proposición:

"En general, parece que asociamos la complejidad con cualquier cosa que encontramos difícil de entender".

De ella se deduce que la complejidad depende directamente de la relación que el observador establece con las cosas. Esto es lo que intenta representar la figura 4. Pero, para poder seguir avanzando, hay que definir primero qué es a lo que llamamos "cosas".

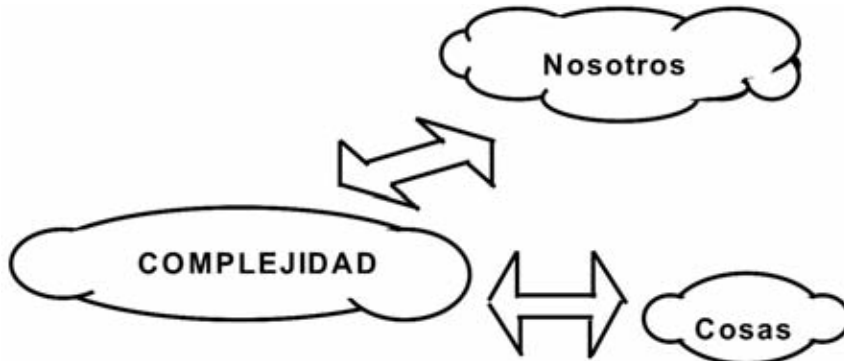


Fig. 4. La complejidad está relacionada tanto con "nosotros" como con las "cosas"

Cuando observamos un objeto cualquiera, lo modelizamos de alguna manera, abstraemos una cierta imagen de él. Las características concretas de esta imagen dependen de nuestros conocimientos, de nuestros intereses particulares, etc. No hay dos personas que contemplando el mismo objeto saquen la misma idea de él. Por estas razones, parece lógico que centremos nuestro estudio en las características de estos modelos realizados por un observador, y no en los objetos "reales". Como dice Klir, no tiene sentido hablar de la complejidad como una propiedad intrínseca de los objetos, sino asociada a modelos, producto de la labor de observación y

abstracción de un observador, modelos que será lo que consideremos como sistemas (hay que destacar el paralelismo del razonamiento de Flood y el de Klir).

El segundo punto en el que hay que detenerse es la caracterización de esta dualidad personas-sistemas, desde el punto de vista de la complejidad. Según Flood, esta caracterización tiene dos caras, las personas y los sistemas (las cosas), y estudia los factores que intervienen en la complejidad de cada uno por separado.

4.1. Complejidad y observador

Para el observador, la complejidad está determinada por tres factores:

- a. Intereses.** A la hora de definir el modelo, el observador estará muy influenciado por los motivos que le mueven en ese momento. El sistema percibido depende en gran manera de los objetivos que persigue el observador. Para ilustrar este punto, Flood cita también el clásico ejemplo de Ashby del neurofisiólogo y el carnicero. Para el primero, el cerebro es algo extremadamente complejo, con su entramado de neuronas, sus complicados procesos bioquímicos... Para él, la transmisión de una descripción detallada del cerebro requeriría mucha información. Pero para el carnicero es algo bastante simple: sólo tiene que distinguirlo entre otros treinta tipos de carne.
- b. Capacidades.** Las capacidades del observador son determinantes respecto a la complejidad del modelo que creará en su mente. Si no tiene ciertos conocimientos, le será muy difícil comprender algunos fenómenos, con lo que para él serán muy complejos. Sin embargo, si los tiene, podrá comprenderlos con mayor facilidad. La complejidad que perciba variará entonces según cómo domine el campo con el que se enfrenta.
- c. Percepciones.** Ya hemos constatado en este mismo capítulo que la forma como ven un mismo objeto dos personas diferentes es diferente. Cada una de ellas hará modelos distintos de la misma sólo por tener una forma distinta de percibir la realidad.

La complejidad y los objetivos del observador

¿Es compleja una silla?. En principio, como ya hemos visto, esta pregunta no tiene sentido si no le asociamos un observador. Comencemos por considerar que este observador es una persona que se quiere sentar. Para ésta, la silla no es más que un mueble con cuatro patas y un respaldo, que conoce desde niño, y que sólo ve como una ayuda para no tener que estar de pie. En resumen, algo muy sencillo.

Pero, sin cambiar de observador, cambiemos el escenario. Esta misma persona quiere construir una silla (por ejemplo, porque es un entusiasta del "hágaselo usted mismo"). Ahora el modelo que se hace de ella es totalmente diferente. Probablemente, se ayude de papel y lápiz, calculará medidas, elegirá el tipo de madera (u otro material) que considere adecuado, etc. La silla, de repente, adquiere una complejidad bastante mayor.

Y no digamos si nuestro sufrido observador es contratado por una empresa de producción de sillas en serie, para que estudie la mejor forma de incluir un nuevo producto en sus cadenas de montaje.

Pues bien, entre los tres escenarios sólo cambia un parámetro: al elaborar el modelo de la silla, el observador está irremediamente condicionado por sus intereses en cada circunstancia. Esto le llevará a concebir modelos de muy diferente grado de complejidad.

4.2. La complejidad en los objetos

Una vez estudiada la parte "humana", tratemos ahora con los "objetos". ¿Qué características tienen en común los modelos que llamamos complejos?. O, dicho de otra forma, ¿cuáles son las causas que provocan que aparezca la complejidad en los objetos?. En una primera aproximación podemos suponer que la complejidad está relacionada con el número de partes y las interrelaciones que forman un sistema, noción que ya recoge Klir en su definición y Flood lleva más allá utilizando para ello un trabajo de Yates [Yates, 1978]:

La complejidad aparece cuando existen una o más de las siguientes características:

- a) Interacciones significativas.
- b) Gran número de partes, grados de libertad o de interacción (multiplicidad).
- c) No linealidad.
- d) Asimetría.
- e) No holonomicidad.

Las dos primeras son las propiedades clásicas, por así decirlo. Las otras tres son propiedades muy interesantes que analizaremos brevemente.

4.2.1. Interacciones significativas

En la naturaleza existen multitud de ejemplos donde un determinado objeto está compuesto de muchos elementos y sin embargo no entra claramente en la categoría de la complejidad. Uno de ellos es el de los gases. Un volumen cualquiera de gas contiene moléculas en un número que es una potencia bastante elevada de 10 (10^{24} para un mol). En razón a este número, un gas sería muy complejo pero no lo es. La razón para ello es que en un gas no hay interacciones significativas. Otra forma de decirlo es que se puede aplicar el principio de superposición de interacciones. En la ciencia que estudiamos se habla de gases "perfectos", adjetivo muy curioso que resalta el hecho de que se está considerando la naturaleza en unas condiciones ideales.

Cuando el objeto en cuestión no es "perfecto" el grado de complejidad reconocible es mayor, incluso mucho mayor. Esto sucede, por ejemplo, al describir el metabolismo de un ser vivo donde cada órgano ejerce una función concreta pero hay tal cúmulo de interdependencias que la descripción del comportamiento global no es trivial en absoluto.

4.2.2. Multiplicidad

Esta es quizá la característica más clásica de un objeto complejo. El simple hecho de que algo esté compuesto por un gran número de partes dificulta su comprensión y su manejo. El ejemplo de los gases tiene un cierto truco. En cierta forma constituye un contraejemplo pero no es así pues el comportamiento del gas como un todo responde a una serie de pautas que son distinguibles en un determinado nivel de resolución. La multiplicidad está directamente relacionada con la variedad. A la escala de una molécula no tiene sentido hablar de presión, volumen o temperatura y esto es así hasta tal punto que las leyes de los gases "perfectos" no mencionan para nada las moléculas. En el sentido del estudio del comportamiento de una mole de granito frente a cambios de temperatura o presión a la misma escala que un gas nadie considera relevante el papel de las moléculas, la única diferencia entre el granito y el gas es el estado.

Podemos retomar el problema del sistema solar para ilustrar la complejidad originada por la multiplicidad. ¿Que hubiera sucedido si en lugar de 10 cuerpos, en tiempos de Newton se hubieran conocido 400 masas planetarias relevantes?. Muy probablemente la mecánica celeste hubiera tardado mucho más en descubrirse. Otro ejemplo son los arabescos, un tipo de decoración con la que se consiguen patrones complicadísimos a base de repetir unas pocas estructuras básicas muy sencillas.

4.2.3. No linealidad

Un sistema no lineal es aquel que no cumple la propiedad de superposición. En el campo de las matemáticas esto significa que si suma una cantidad a la entrada, la salida no se modifica en esa misma cantidad. Estos sistemas son bastante más difíciles de comprender que los lineales. Gran parte de las matemáticas que se emplean en la física, por ejemplo, están encaminadas a evitar todo aquello que "huela" a no lineal. Con todos los avances en matemáticas hechos hasta ahora todavía somos incapaces, salvo poquísimas excepciones, de resolver no linealidades.

Una rama de la ciencia muy popular actualmente es el estudio del caos. El caos, un comportamiento aparentemente aleatorio, procede de no linealidades. Un ejemplo muy ilustrativo de lo que realmente representa encontrarse con una no linealidad es la siguiente ecuación:

$$Y = r * X * (1-X)$$

Esta ecuación es muy utilizada en el estudio del crecimiento de todo tipo de poblaciones de seres vivos. X representa la población en un momento dado e Y la población en el momento siguiente, r es un parámetro que se calcula en función de las condiciones del entorno. El término 1-X compensa a X para que la ecuación tienda a estabilizarse (sin él, sería creciente). Pues bien, resulta que, para una población inicial dada, según el valor de r ese sistema se estabiliza u oscila con período 2, 4, 8, 16, 32 ... o aparece el caos. Ya no es que el resultado sea mayor o menor, es que cambia la propia esencia de ese resultado. Una característica típica de las no linealidades, diferentes condiciones de partida, conduce a resultados totalmente diferentes y generalmente contra-intuitivos.

4.2.4. Asimetría

La asimetría significa diferencia. Cuando un proceso es asimétrico las propiedades aplicables al principio no son las mismas que las que hay al final del mismo. Un árbol binario simétrico (como el de la figura 5) es sencillo de construir, siguiendo un procedimiento recursivo:

1. Partir de un nodo inicial.
2. Sustituir cada nodo terminal por una bifurcación terminada en otros dos nodos.
3. Volver al punto 2.

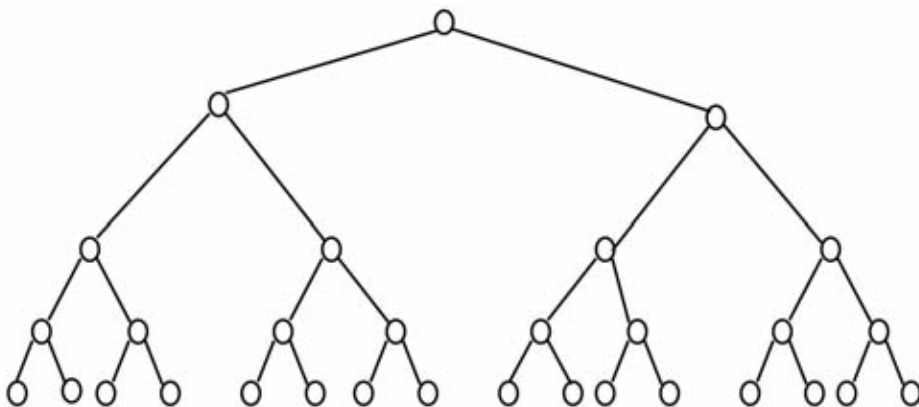


Fig. 5. Árbol binario simétrico

Si introducimos la más pequeña asimetría en ese árbol la descripción se complica mucho. Hay que especificar cada nivel y cada nodo en el que se produce una asimetría. La descripción es más compleja a medida que se introducen más asimetrías. Como ejercicio, trate el lector de establecer un algoritmo para construir el árbol de la figura 6. Y obsérvese cómo varía el algoritmo si introdujésemos aún más asimetrías. Lo mismo sucede en la teoría de grafos y redes. Para estructuras más o menos simétricas existe una teoría matemática completa y bien desarrollada. Cuando el grafo o la red se hace asimétrico, como sucede en la mayoría de los casos, las teorías son mucho más difíciles de formular y son incompletas.

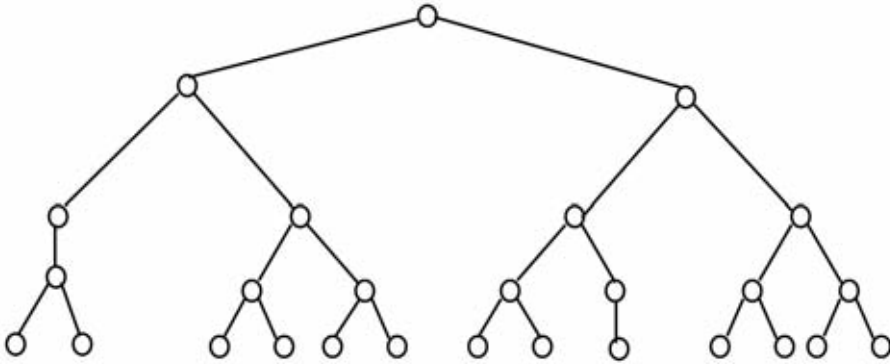


Fig. 6. Árbol con asimetrías

4.2.5. No holonomicidad

A pesar de la palabreja este concepto tiene un significado muy claro. Se refiere a las leyes (nomic) y al todo (holos). Cuando un conjunto se comporta de acuerdo con leyes de validez general hay una integridad, una cierta seguridad sobre qué va a pasar. Pero si, por el contrario, hay partes del todo que pueden quedar temporalmente "excluidas" de esas reglas y comportarse de acuerdo con patrones ajenos al conjunto aparecen estados transitorios de anarquía y, en consecuencia, una mayor complejidad. Este es un fenómeno común, por ejemplo, en electrónica digital: por circunstancias desconocidas aumenta la tensión en un punto y un bit cambia de estado repentinamente, esto puede ser un fallo leve o puede ser gravísimo según el bit afectado. A la hora de diseñar el sistema se atiende desde luego a esas reglas generales, al todo, pero también se prevén fallos de este tipo a través de redundancias y controles exhaustivos.

A partir de todo ello, Flood establece el marco conceptual que propone para definir la complejidad, este marco es el que se representa en la figura 7:

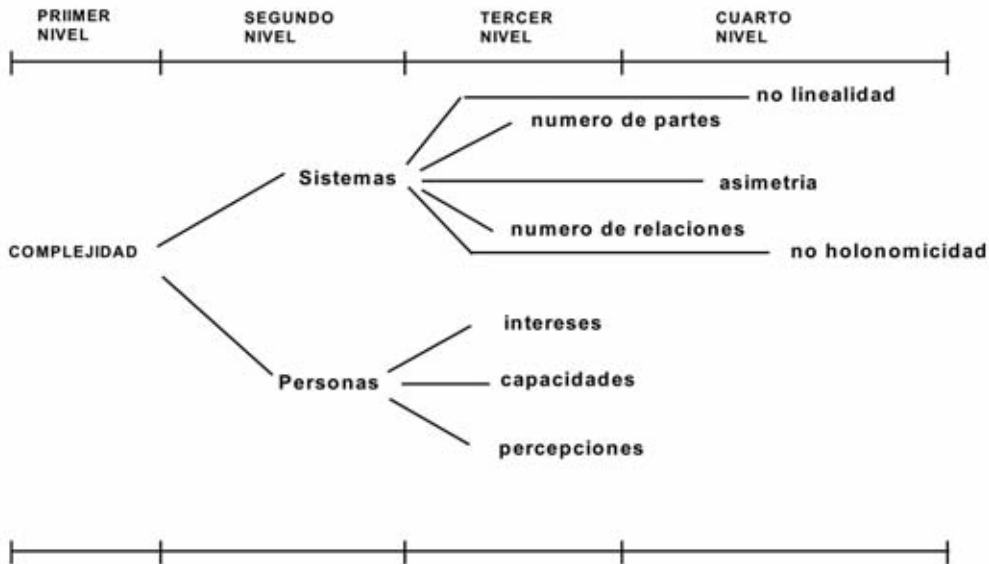


Fig. 7. Desensamblado de la complejidad, según Flood

A modo de resumen, podemos decir que la complejidad es para Flood el resultado de la interacción entre las personas y las cosas y, por tanto, ha de ser descrita en función de esos dos elementos. Las personas influyen en la complejidad porque son los observadores que van a definir el sistema relevante sobre el que se va a percibir la complejidad y de ahí que los factores que introducen sean los intereses, las capacidades y las percepciones. En el lado de los sistemas, una vez establecidos por el observador, la complejidad va a estar influenciada en un primer término por lo que se ha venido considerando como factores clásicos de la complejidad: el número de partes y el número de relaciones. Estos dos factores forman un nivel estructural de la complejidad, en el sentido de que establecen una complejidad estática relativa a la organización del sistema. Pero no son los únicos factores a considerar ya que también hay una complejidad asociada al comportamiento y determinada por la asimetría, la no holonomicidad y la no linealidad.

El conjunto es una imagen muy completa de lo que es la complejidad y de los factores asociados a ella, que sirve, como veremos a continuación, de punto de partida para establecer lo que Flood llama una nueva dimensión de la complejidad.

4.3. Una nueva dimensión de la complejidad

Checkland [Checkland, 1981] propone una clasificación de las ciencias desde la óptica de la complejidad, basada en la clásica distinción de Comte:

FÍSICA-QUÍMICA-BIOLOGÍA-PSICOLOGÍA-SOCIOLOGÍA

Esta clasificación es llamada en la literatura inglesa el "espectro hard-soft". Se considera que hacia la izquierda, comenzando por la Física, tenemos la "ciencia básica", con los conceptos primarios de energía, masa, movimiento... Desplazándonos hacia la derecha nos encontramos con disciplinas que se ocupan de campos cada vez más complejos. Así, las reacciones químicas, aunque se pueden explicar en cierta medida en función de fenómenos físicos, son "algo más" que eso. Cuando observamos que de la mezcla de hidrógeno y oxígeno, bajo ciertas condiciones, se obtiene agua, parece que estamos ante un fenómeno de un nuevo orden que "trasciende" la Física: emergen nuevos conceptos, aumenta la complejidad. Igual ocurre con la Biología respecto a la Química. Basta con pensar en el crecimiento de una planta a partir de una semilla, o de un animal a partir de una única célula. Y podemos seguir con la psicología y la sociología.

En todos estos casos se observa que los fenómenos que ocupan alguna parte de la secuencia no pueden ser explicados totalmente en términos de las ciencias que están más a su izquierda. Siempre hace falta introducir nuevos conceptos. Es cierto que la vida se basa en un conjunto de reacciones químicas. Pero este conjunto es tan vasto, y las relaciones dentro de él tan intrincadas, que aunque conozcamos con precisión el desarrollo de todas las reacciones, si no "cambiamos de escala" siempre se nos escapará la imagen total de lo que ocurre. Es algo parecido a un puzzle: cada pieza representa una porción de la imagen total, pero aunque las examinemos detenidamente por separado, no tenemos una visión completa hasta que no lo montamos entero y nos separamos un poco para contemplarlo.

Esta representación "tradicional" del espectro de las ciencias es claramente unidimensional. Sin embargo, cuando nos movemos en zonas donde el humano es parte estudiada, parecen emerger nuevas dificultades. Estas pueden estar basadas en la "innata tendencia de los seres humanos a apreciar la 'misma' situación desde diferentes puntos de vista" [Flood, 1987]. Los motivos según los cuales actúan las personas son muy diferentes, y difíciles (si no imposibles) de predecir. Debido a esto parece que hemos encontrado un salto cualitativo en el tipo de complejidad con que nos encontramos en estos casos. Podemos plantearnos por tanto una nueva dimensión de complejidad, que se comenzaría a mostrar en disciplinas como la ecología, para hacerse clara con la psicología y sobre todo con las ciencias sociales.

Flood relaciona esta observación con los tres rangos de complejidad descritos por Weaver, estableciendo el esquema que se muestra en la figura 8.

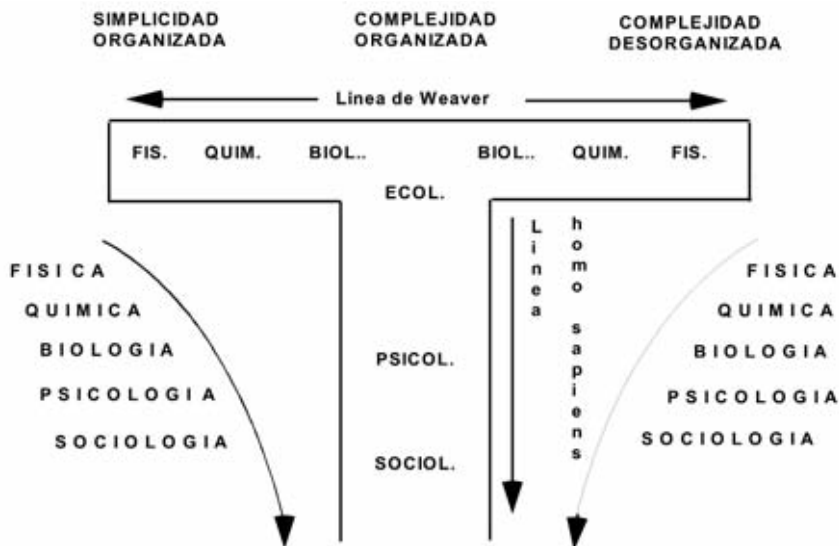


Fig. 8. Representación gráfica de la T de Flood

En la parte horizontal de la T tenemos los tres rangos de complejidad de Weaver. Comenzando por la izquierda, los fenómenos de simplicidad organizada suelen ser tratados por la física (pocos elementos que se comportan determinísticamente), y en menor medida por la química. Parte de esta última, y más claramente, la biología, marcan la zona de transición hacia los problemas de complejidad organizada (aumenta el número de elementos, y la aleatoriedad, pero no demasiado). Y siguiendo hacia la complejidad desorganizada nos encontramos de nuevo con la química y, en el extremo, con la física (por ejemplo, la mecánica estadística). A esta línea la llama Flood "línea de Weaver".

Mientras tanto, en la parte vertical, dentro del ámbito de la complejidad organizada, nos movemos en la "línea homo sapiens". Cuanto más hacia abajo dentro de ella, más marcados están los sistemas por la actividad humana, con esa diferencia cualitativa de la que ya hemos hablado. A medida que nos movemos por esta línea vertical, los problemas se van transformando en "messiness", algo que es "más que un problema", debido a los diferentes valores, intenciones, etc, que introducimos las personas en los sistemas.

Esta es la nueva dimensión de la complejidad que propone Flood, una complejidad que está más allá de la considerada tradicionalmente por la ciencia y que tiene su origen en los sistemas de actividades humanas, en la línea "homo-sapiens", en la que el humano es el sujeto principal y fuente indiscutible de complejidad.

5. F. Sáez Vacas. Modelo de tres niveles de complejidad

5.1. Complejidad. Definición por extensión

Ante la dificultad de definir una idea se pueden escoger varias soluciones. Una de ellas es intentar dar una panorámica lo más amplia posible de su significado, estableciendo una especie de catálogo de las situaciones que conducen a ella. Este es el objetivo de la definición amplia de complejidad de Sáez Vacas [Sáez Vacas, 1990]:

Complejidad es el nombre que damos a la condición de los seres humanos, objetos, fenómenos, procesos, conceptos y sentimientos cuando cumplen uno o varios de estos requisitos:

- A. Son difíciles de entender o explicar.
- B. Sus causas, efectos o estructura son desconocidos.
- C. Requieren una gran cantidad de información, tiempo o energía para ser descritos o gestionados, o un esfuerzo muy amplio y coordinado por parte de personas, equipos o maquinaria.
- D. Están sujetas a varias percepciones, interpretaciones, reacciones o aplicaciones que, frecuentemente, son contradictorias o desconcertantes.
- E. Producen efectos deseados y no deseados (o difíciles de controlar).
- F. Su comportamiento, dependiendo del caso, puede ser impredecible, extremadamente variable o contraintuitivo.

Esta definición de complejidad intenta presentarla como un concepto amplio que abarca muchos otros. Cada uno de los puntos que recoge la definición trata un aspecto diferente de la complejidad, a saber:

1. Epistemología de la complejidad. A partir de un objeto complejo es más difícil obtener un modelo que recoja todas sus características y sea representativo de su comportamiento. Este punto está directamente relacionado con el punto D, que puede considerarse como una consecuencia de éste.

2. Estructura y complejidad. Más que desconocerse totalmente, lo que sucede en realidad es que en un objeto o sistema complejo no se pueden llegar a conocer todos los factores que influyen en su comportamiento o en qué medida lo hacen (causas), de ahí que tampoco se conozcan los efectos que producen. En cuanto a la estructura, la existencia de múltiples interrelaciones entre los elementos o la multiplicidad de elementos imposibilita obtener una imagen completa de la organización interna del sistema.
3. Información y complejidad. La multiplicidad de elementos característica de los sistemas u objetos complejos implica un crecimiento exponencial en el número de relaciones y por tanto en la cantidad de información requerida para describir el sistema. Se puede pensar en utilizar la teoría de la información (como propone Ashby) para intentar desarrollar un tratamiento formal de la complejidad.
4. Ambigüedad y complejidad. La dificultad para establecer un criterio válido de interpretación de un objeto o sistema complejo implica que va a haber diferentes formas de verlo. Este punto resalta la importancia del observador.
5. Complejidad y control. Relacionado con el punto 2. En los sistemas complejos es frecuente no poder optimizarlos sino limitarse a soluciones de compromiso en las que se intentan equilibrar distintos efectos. Esto implica una complejidad adicional pues indica que el sistema puede, en un momento dado, comportarse de forma opuesta a lo que sería deseable.
6. Predicción y complejidad. Una consecuencia de los puntos 2 y 5. Dado que no conocemos exactamente la estructura de un sistema complejo y tampoco podemos asegurar que se controlan todos los parámetros que sobre él influyen, el comportamiento será, en consecuencia, sorprendente cuando entre en juego uno de los factores que no se han considerado.

5.2. Complejidad y sistemas artificiales

Los trabajos sobre complejidad de Sáez Vacas están básicamente orientados a su aplicación en la informática y en las tecnologías de la información. Estas, como sistemas artificiales que son, presentan una problemática distinta a la que se deriva de los sistemas naturales. El modelo de tres niveles del que nos ocuparemos más adelante refleja este sesgo intencionado en la interpretación de Sáez Vacas, para quien el problema del estudio de la complejidad en los sistemas artificiales se puede articular sobre cinco preguntas:

- A. ¿De qué factores depende?
- B. ¿En qué formas se encuentra?

- C. ¿Cuáles son sus consecuencias?
- D. ¿Cómo evoluciona?
- E. ¿Qué se puede hacer para medirla o gestionarla?
- A. Es decir, ¿cómo surge la complejidad?, ¿es posible distinguir una serie de elementos o causas cuyo efecto inmediato es la aparición de la complejidad?. Es importante conocer cómo se presenta la complejidad y cuál es su origen para poder tratarla.
- B. ¿Es posible encontrar diferentes categorías o tipos de complejidad? En el caso de disponer de un mapa de los tipos de complejidad existentes también podríamos pensar en una clasificación de los métodos de tratamiento de la complejidad facilitando así la tarea de tratarla. El primer paso sería establecer si existen efectivamente diferentes categorías o si, por el contrario, se puede hablar de una complejidad única y genérica que aparece en todo aquello que calificamos como complejo. El modelo de 3 niveles establece este tipo de categorías de complejidad.
- C. En este caso, ¿cuáles son las consecuencias de la complejidad en las tecnologías de la información o, en general, en cualquier sistema artificial. Para poder tratar un problema debemos conocer cuáles son sus consecuencias y los efectos que de él se derivan, para ello es fundamental poder definirlo con claridad (punto A) y, si es posible, categorizarlo dentro de una serie de problemas reconocibles (punto B).
- D. Un problema muy interesante, sobre todo en las tecnologías de la información, pues, como Sáez Vacas demuestra, la complejidad no es estática sino que es el producto de una dinámica de interacción entre diversos conceptos. Una dinámica que hay que conocer para poder tratar la complejidad a largo plazo.
- E. Y, finalmente, un punto controvertido en cuanto a la complejidad. Existen muchas teorías al respecto y casi nadie se pone de acuerdo en cómo se puede medir la complejidad y lo mismo se puede decir de cómo gestionarla. Prácticamente cada autor propone un método o un modelo para ello, métodos y modelos que son más o menos válidos dentro del campo de interés para el que están pensados.

Estas preguntas trazan todo un campo de investigación dentro del estudio de la complejidad, con el objetivo de buscar una cierta unidad a la interpretación de diferentes tipos de complejidad, entre los cuales cabe citar a los siguientes, que, de una u otra forma, tendrían relación con las tecnologías de la información:

- a. Complejidad del sistema.
- b. Complejidad del programa.
- c. Complejidad computacional.
- d. Complejidad epistemológica.
- e. Complejidad organizativa.
- f. Complejidad interpretativa ("complex thinking").

El programa de esta investigación consiste en poner de manifiesto área por área (software, arquitectura de ordenadores, redes, etc.) cómo se ha respondido y cómo se puede responder a algunas (o a todas) de esas cinco preguntas.

5.3. Modelo de tres niveles

Esta teoría fue propuesta inicialmente para su aplicación a la informática y su propósito era categorizar la complejidad que caracteriza a los entornos de este tipo. Si extraemos de esa teoría todo lo relacionado con la informática, nos quedamos con la esencia del razonamiento, que es bastante independiente del campo de aplicación. Y muy ilustrativa de cómo se puede interpretar la complejidad [Sáez Vacas, 1983].

Su autor distingue tres niveles de complejidad. El primero de ellos abarca la complejidad de los objetos aislados; en el caso de la informática, es la complejidad de circuitos, algoritmos, programas, etc, tratados como objetos separados de otros. Este es el tipo de complejidad con que se enfrentan todos los especialistas en estas tecnologías y es, en general, la que percibe todo el mundo. En un sentido más amplio, ésta es la complejidad de cada uno de los diferentes elementos que componen un sistema, considerados en sí mismos y no como referencias interiores de un todo. Evidentemente, ellos a su vez podrían ser enfocados como un todo, puesto que pueden estar formados por otros elementos, hecho del que nos ocuparemos un poco en breve.

El segundo nivel de complejidad aparece porque, en general, y en la informática en particular, los objetos nunca están aislados, sino que forman un grupo de elementos interconectados con un determinado objetivo. Surge aquí la noción de

sistema que lleva aparejada una complejidad diferente y de orden superior a la del primer nivel y que es por lo habitual de naturaleza **metadisciplinar**, en el sentido de estar situada más allá de la mera especialización. La llamaremos complejidad sistémica. En informática los ejemplos son numerosos: un sistema operativo, una red de ordenadores, un ordenador aislado o incluso un circuito integrado VLSI, etc. Es un nuevo nivel de complejidad, porque ya no se está tratando con un simple conglomerado de objetos, sino que de la unión de éstos emerge un conjunto de propiedades diferente a la mera suma de las propiedades de sus componentes. En concreto, surge una serie de interrelaciones que antes no existían o que no interesaban, pero que ahora son fundamentales para definir el comportamiento del grupo.

Finalmente, el tercer nivel de complejidad surge de la interacción (a veces choque) de los sistemas tecnológicos y los sistemas sociales, dando lugar a la complejidad antropotécnica, que se manifiesta específicamente por fenómenos relacionados con el desorden, la incertidumbre, la desorganización, la inestabilidad, la entropía, la borrosidad..., algo que está a la orden del día en la informática práctica. Es una clase de complejidad de carácter **multidisciplinar** y metatecnológico.

En nuestro estudio este nivel es de primordial importancia pues es aquí donde se expresa toda la problemática de las interfaces, de la aceptación de la tecnología, de la adaptación humana, etc. La formulación de este nivel genera un salto cualitativo importante en la concepción de la complejidad, pues ahora no se limita su percepción al aspecto puramente técnico, hasta cierto punto formalizable y estructurable, sino que se introduce de lleno la perspectiva humanística, aspecto importantísimo de la tecnología. Entre otras cosas, pone de manifiesto la importancia del observador, un agente activo, fundamental, parte del propio sistema, que interactúa y evoluciona con él.

Por otro lado, como ya se ha mencionado, un objeto puede considerarse desde varios puntos de vista. Un elemento puede considerarse como tal, si lo que interesa es un sistema mayor del cual forma parte, en este caso estaremos ocupándonos de la complejidad del segundo nivel, o puede considerarse como un sistema en sí mismo, si lo estudiamos como un objeto aislado, en cuyo caso estaremos ocupándonos de la complejidad del primer nivel. Todo dependerá de nuestro punto de vista y de nuestro interés en cada momento. Veamos un ejemplo.

Un ordenador personal puede verse como un objeto de complejidad grande, compuesto por muchos elementos y dedicado a una aplicación concreta. En este caso, el ordenador se estudia como un objeto cuya complejidad está en el segundo nivel, el ordenador es un sistema, sus propiedades de complejidad han de buscarse en la clase de complejidad sistémica. Las diferentes partes de que se compone el ordenador, tarjetas de procesamiento, de memoria y gráficas, discos de almacenamiento, fuente de alimentación, monitor, etc., son objetos cuya complejidad está en el primer nivel, donde se pueden estudiar por separado, aisladamente.

Es importante comprender que, en la versión anterior, el ordenador personal se considera como un sistema formado por muchos componentes y su complejidad emerge a partir de una determinada interconexión de las complejidades del primer nivel de sus componentes. Por el contrario, en una red de ordenadores, el ordenador pasa a ser un objeto de primer nivel, su complejidad interesa ahora considerarla como la de un objeto aislado, mientras que el sistema, la red, pasa a definir la complejidad de segundo nivel, la complejidad que surge al considerar la conexión e interrelación de diversos objetos 'simples', en este caso, ordenadores. Considerar el ordenador en un nivel o en otro depende, como se ve, del punto de vista que adoptemos, el de diseñador de redes de comunicación, en cuyo caso manejaremos ordenadores como elementos, o el del diseñador de ordenadores personales, en cuyo caso el ordenador pasará a ser el sistema de orden superior que se intenta construir a partir de una serie de elementos.

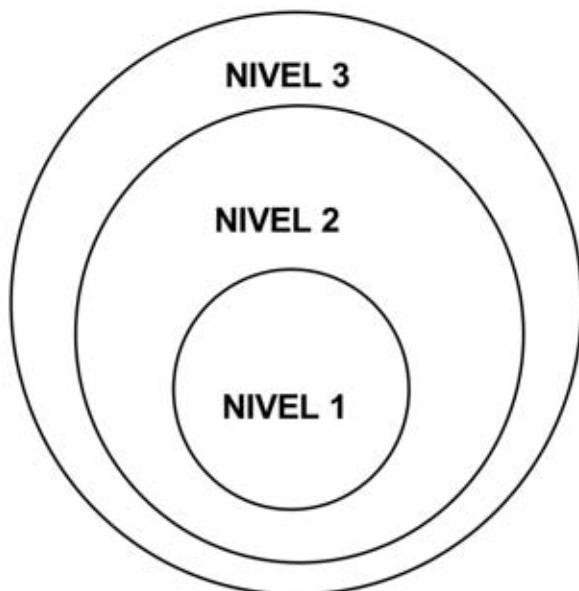


Fig. 9. Modelo de tres niveles de complejidad (F. Sáez Vacas, 1983)

Por último, hay que destacar que a un nivel superior también existe la complejidad del inferior. Esto en el ejemplo anterior se ve muy claramente, una red presenta una complejidad característica, de orden superior a la de los ordenadores que la forman, pero al mismo tiempo también contiene la complejidad asociada con esos ordenadores. Esto es lógico si consideramos que un sistema está formado por sus elementos y las relaciones entre ellos.

En términos cibernéticos, dejando la informática de lado y a modo de resumen, lo que se plantea en este modelo de complejidad son tres niveles distintos de la misma, dependientes, como siempre, del observador. Para cada nivel, las consideraciones, técnicas y conceptos relevantes son diferentes y específicos, si bien, desde un punto de vista constructivo (partiendo desde la tecnología) todo nivel está incluido en el superior.

Complejidad de los objetos aislados: **Complejidad Individual.**

Complejidad de los objetos interconectados: **Complejidad Sistémica.**

Complejidad de la interacción tecnología-sociedad: **Complejidad Antropotécnica.**

Si nos fijamos, el nivel de complejidad antropotécnica es el equivalente en el campo tecnológico al concepto de complejidad línea homo-sapiens, elaborada años después por Flood [Flood, 1987].

El modelo de los tres niveles representa un enfoque sistémico ascendente en cuanto que el punto de partida constructivo del sistema es la tecnología. Sin embargo, en lo que se refiere a sus objetivos de diseño o verificación es descendente (top-down): el tercer nivel incluye al segundo y éste, al primero.

6. Resumen

La **clasificación de los sistemas** según Weaver se hace en función de su **número de elementos** y de su grado de **aleatoriedad**. Estos criterios dan lugar a tres rangos de complejidad: **simplicidad organizada, complejidad desorganizada y complejidad organizada.**

Klir relaciona la complejidad de un objeto con el **número de partes** que lo componen y de **interrelaciones** entre esas partes. También se menciona la importancia del **observador**. De la interacción entre el observador y el objeto surge la idea de sistema: la imagen que del objeto tiene el observador. De aquí la importancia de establecer un marco conceptual para tratar con los sistemas, que se concreta en la **jerarquía epistemológica** que propone: sistema-fuente, sistema de datos, sistema-generativo, sistema-estructura, metasistema,... También se enuncian dos principios generales de la complejidad de los sistemas: la relación de la complejidad con la **información que describe** el sistema, y con la necesaria para resolver su **incertidumbre.**

Flood, partiendo también de la relación de la complejidad con el **objeto** y con el **observador**, analiza los factores que en cada uno de estos dos elementos están relacionados con ella. Para el observador serán los **intereses**, las **capacidades** y

las **percepciones**, y para los sistemas el **número de partes**, de **relaciones** (ambas ya mencionadas por Klir), la **no linealidad, asimetría y no holonomicidad**. También clasifica las ciencias de una forma que extiende la clasificación de sistemas de Weaver al introducir la **línea homo-sapiens (T de Flood)**.

Sáez Vacas, intentando dar una visión general del campo que abarca la complejidad, aporta su definición, que supera el concepto de complejidad descriptiva de un objeto, para abarcar toda clase de operaciones o **situaciones de complejidad** con un objeto por parte de toda clase de observadores. También elabora un **modelo en tres niveles**, a saber: complejidad **individual**, complejidad **sistémica** y complejidad **antropotécnica**. Y distingue, ya dentro del campo de las tecnologías de la información, seis **tipos de complejidad**: del **sistema**, del **programa**, **computacional**, **epistemológica**, **organizativa e interpretativa**.

7. Bibliografía

Dividida en dos apartados. En Notas Bibliográficas se comentan aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

El artículo de Klir "**Complexity: some general observations**" [Klir, 1985], ofrece, entre otras cosas, unas interesantes consideraciones sobre el papel del observador y el objeto en la complejidad, una descripción de la jerarquía epistemológica de tipos de sistemas, unas notas sobre los tres rangos de complejidad de Weaver y un estudio sobre la complejidad computacional.

Flood, en "**Complexity: a definition by construction of a conceptual framework**" [Flood, 1987], desarrolla un marco conceptual donde encajar diferentes aspectos relacionados con la complejidad y expone su ampliación de la "línea de Weaver" (la "T de Flood").

En "**Facing informatics via three level complexity views**" [Sáez Vacas, 1983], expone Sáez Vacas su modelo de tres niveles de complejidad aplicándolo esencialmente en el campo de las tecnologías de la información.

Referencias bibliográficas

Checkland, P.B. (1981): "**Systems thinking, systems practice**", John Wiley, Nueva York.

Klir, G.J. (1985): "Complexity: Some General Observations", **Systems Research**, vol.2, núm.2, pp.131-140.

Flood, R.L. (1987): "Complexity: a definition by constructing a conceptual framework", **Systems Research**, vol.4, núm.3, pp.177-185.

Orchard, R.A. (1972): "On an approach to general systems theory", editado en **Trends in general systems theory**, John Wiley & Sons, Nueva York.

Pippenger, N. (1978): "Complexity theory", **Scientific American**, 238, 6, 1978, pp.90-100.

Sáez Vacas, F. (1983): "Facing informatics via a three level complexity view", **X international Congress on Cybernetics**, Namur, Bélgica, pp.30-40.

Sáez Vacas, F. (1987): "**Towards a conceptual remodeling of information technologies based on a broad consideration of complexity**", 31st Annual Meeting of the International Society for General Systems Research, Budapest, 1-5 junio.

Sáez Vacas, F. (1990): "**Ofimática compleja**", Fundesco, Madrid.

Wagensberg, J. (1985): "**Ideas sobre la complejidad del mundo**", Tusquets, Barcelona.

Weaver, W. (1948): "Science and complexity", **American Scientist**, núm.36, pp.536-544.

Yates, F.E. (1978): "Complexity and the limits to knowledge", **American J. Physiol.**, núm.4, pp.201-204.

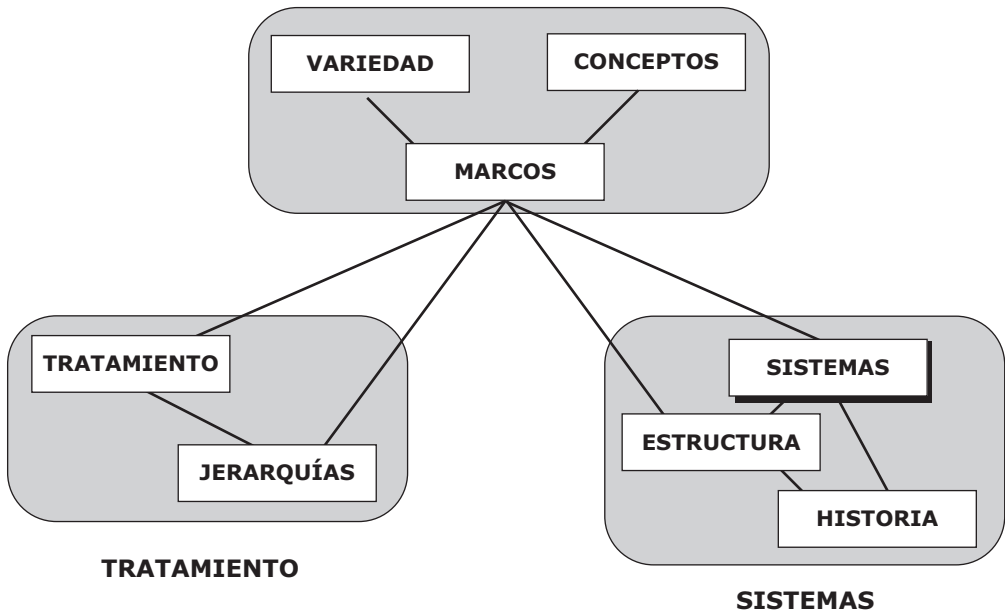
Capítulo 4: Complejidad y sistemas

"La ciencia ha explorado el macro y el microcosmos, tenemos una idea más o menos clara de la disposición de la Tierra. La gran frontera sin explorar es la complejidad. Los sistemas complejos incluyen el cuerpo y sus órganos, especialmente el cerebro, la economía, los sistemas de población y evolución, el comportamiento animal, las grandes moléculas, etc... Algunos de estos objetos se pueden simular con un ordenador y se pueden modelar de forma bastante precisa; otros no pueden simularse de forma más sencilla que el sistema mismo. Los científicos, en este nuevo esfuerzo interdisciplinar, han comenzado a enfrentarse a los sistemas complejos y, de forma destacada, están entendiendo cómo la complejidad puede surgir de la simplicidad. Por ejemplo, los autómatas celulares, un conjunto artificial de puntos en una pantalla que se organizan a sí mismos de acuerdo con reglas definidas y simples, son ejemplos de un comportamiento complejo que surge de la simplicidad. ... Todo lo que existe puede verse como sistemas complejos construidos a base de componentes simples".

H.R. Pagels [1989, p. 12]



ENFOQUES Y TEORÍAS



1. Introducción

La relación entre la complejidad y la idea de sistema va a ser una constante a lo largo de todos estos apuntes. La máxima "el todo es más que la suma de las partes", atribuida a Aristóteles, reconoce de forma explícita la existencia de una complejidad asociada al conjunto que no es directamente extrapolable de los elementos que lo forman, por lo tanto, el reconocimiento de la existencia de un sistema en un determinado objeto lleva aparejado el hecho de la complejidad producida por la interacción múltiple entre los elementos que lo forman.

Pero, ¿qué es un sistema?, la definición no es fácil y depende en gran medida de lo que se pretenda hacer con el sistema (consultar al respecto el capítulo sobre la **Visión Estructural y Funcional de los Sistemas**). A grandes rasgos, distinguir un sistema es reconocer la existencia de un conjunto de elementos que interactúan entre sí definiendo en esa interacción unos objetivos o un comportamiento globales que no son directamente deducibles de los elementos aislados. Esto significa que el estudio de un sistema no se puede hacer limitándose a estudiar los elementos que lo forman, sino que hay que tomar el "todo" como referencia.

En el campo de los sistemas se ha trabajado mucho intentando clasificar los diferentes tipos de "todos" que se pueden encontrar, buscando criterios que indiquen cómo se deben considerar los sistemas. Nuestra idea es partir de una de estas clasificaciones y con un análisis cuidadoso se puede establecer qué tipo de complejidad presenta cada uno de los niveles de esa clasificación y, de esta forma, profundizar en el significado de la complejidad.

2. De la célula al Sistema Solar

Si únicamente nos valemos de la noción genérica de la complejidad es difícil contestar a esta pregunta: ¿qué es más complejo, el sistema solar o una célula?. De acuerdo con las definiciones de complejidad que habitualmente se manejan, esta cuestión es prácticamente irresoluble. Ambos, sistema solar y célula, son difíciles de entender, requiere un gran esfuerzo describirlos por completo, hay diferentes formas de verlos e interpretarlos, etc. Pero tenemos una idea intuitiva de cómo pueden ser las cosas: el sistema solar está compuesto de cientos de miles de cuerpos, sin duda muchos más de los que componen la célula. El sistema solar tiene un radio aproximado de unos 12.000 millones de kilómetros y es cerca de 20 órdenes de magnitud más grande que una célula. Ha existido desde mucho antes que cualquier célula conocida y lo seguirá haciendo cuando éstas hayan desaparecido de la tierra. Y, a pesar de todo ello, a efectos prácticos, una célula es más compleja que el sistema solar. Podría argüirse que si se intenta describir el sistema solar con el nivel de resolución de la célula éste resultaría mucho más complejo, es cierto, pero no es un problema real, no tiene ningún sentido, de igual forma que no tiene sentido estudiar la célula a la escala del sistema solar.

Se dice que hoy en día es más fácil llevar un hombre a la Luna que comprender por qué una célula crece o no crece. ¿Por qué?. Ha de existir una respuesta desde el punto de vista de la complejidad. ¿Qué propiedades o características tiene la célula que la hacen más compleja que el sistema solar?. Una de ellas es definitiva: que está viva. Una primera conclusión sería pensar que todo organismo vivo es más complejo que uno inanimado, estableciendo así una frontera entre dos niveles de complejidad, la complejidad de la materia viva y la complejidad de la materia inerte (por supuesto, considerando el sistema solar como soporte de la vida y sin tomar en consideración los elementos vivos que lo componen). El siguiente paso sería establecer diferentes categorías dentro de estos dos niveles atendiendo a algún criterio que nos indique donde hay más o menos complejidad. Es decir, intentaríamos establecer una serie de factores que cuando aparecen indican complejidad. Uno de ellos podría ser el hecho de estar vivo. Este tipo de razonamiento es el que se ha hecho frecuentemente para identificar distintas clases de sistemas, por eso nos vamos a tomar el estudio de los sistemas como base para profundizar en la complejidad.

3. Sistemas y Complejidad

Una de las primeras clasificaciones sobre sistemas es la que proporcionó Boulding en 1956, en un trabajo ya clásico de Teoría de Sistemas, cuyo título era: "Teoría General de Sistemas. El esqueleto de la Ciencia" (Boulding fue uno de los promotores de la Sociedad Internacional para la Investigación de los Sistemas Generales) (ver la **Historia del Enfoque Sistémico**). Esta clasificación aparece en el cuadro adjunto (extraída de [Checkland, 1981]).

Para mejor comprender esta clasificación hay una serie de ideas que se deben tener en cuenta. La primera de ellas es que las propiedades que caracterizan a un nivel aparecen con ese nivel, es decir, no aparecen en niveles inferiores y se mantienen en niveles superiores. La segunda es que la complejidad aumenta a medida que se sube de nivel (asumiendo el 1 como el inferior). Es decir, para un observador exterior se hace progresivamente más difícil predecir el comportamiento del sistema, que depende cada vez más de decisiones no programadas. Finalmente, los sistemas de nivel inferior están incluidos en los de nivel superior, por ejemplo, los hombres presentan todas las características de los niveles inferiores (del 1 al 6), además de las nuevas propiedades que hace que se les considere como un nuevo nivel.

Boulding, cuando publicó este trabajo, intentaba evitar la generalidad absoluta "todo lo que podemos decir acerca de prácticamente todo es casi nada", y para ello buscó situarse en un nivel intermedio entre "lo muy específico, que no significa nada, y lo general, que no tiene contenido". Históricamente, el nivel 2 representa el área de interés de las teorías de organización debidas a Taylor y el nivel 3 representa el campo de la cibernética, al menos en sus inicios después del libro de

Wiener y cuando nació la teoría de control. Los niveles superiores son el campo de estudio de las teorías de dirección, teorías sociales, psicología, etc.

NIVEL	CARACTERÍSTICAS	EJEMPLOS CONCRETOS O ABSTRACTOS	DISCIPLINAS RELEVANTES
1. Estructuras	Estático	Estructuras cristalinas puentes	Descripción verbal o gráfica en cualquier disciplina
2. Máquinas	Movimiento predeterminado	Relojes, Máquinas, Sistema Solar	Física, Ciencias Naturales clásicas
3. Mecanismos de control	Control en bucle cerrado	Termostatos, Mecanismos homeostáticos	Teoría de control, cibernética en organismos
4. Sistemas abiertos	Estructuralmente automantenidos	Células biológicas (teoría de la información)	Teoría del metabolismo
5. Organismos inferiores	Todos organizados con partes funcionales, crecimiento y reproducción	Plantas	Botánica
6. Animales	Un cerebro gestor del comportamiento, habilidad para aprender	Animales en general	Zoología
7. Hombre	Conciencia de sí mismo, conocimiento sobre el conocimiento, lenguaje simbólico	Seres humanos	Biología, Sociología
8. Sistemas socio-culturales	Roles sociales, comunicación, transmisión de valores	Familias, Boy Scouts, clubs de bebedores, naciones	Historia, Sociología Antropología, Ciencia del Comportamiento
9. Sistemas Trascendentes	Conocimiento Inaprehensible	La idea de Dios	Teología

(A partir de Boulding, 1956 Citado por Checkland, 1981)

La clasificación de Boulding presenta el problema de que la distribución de complejidad entre los niveles no es uniforme, es decir, el salto para pasar de un nivel a otro no es siempre del mismo orden de magnitud. Como señala Checkland, la jerarquía es intuitivamente correcta, pero mientras que para los niveles inferiores se puede establecer una relación con una serie de parámetros que los determinan (estructuras = masa y tamaño; máquinas = masa, tamaño y tiempo; mecanismos de control = masa, tamaño, tiempo e información) esta relación es muy difícil de establecer para organismos superiores. En cualquier caso, es una jerarquía muy expresiva de las diferencias existentes entre los sistemas que se mencionan y de la diferencia de complejidad entre cada uno de ellos. En esta clasificación se puede establecer una línea divisoria muy clara definida por la propiedad de estar vivo. Esto establece una distinción importante en la complejidad como ya hemos mencionado. También se puede trazar otra división entre los sistemas humanos y los que no lo son, pues en los primeros aparece una complejidad radicalmente diferente a la que se puede distinguir en el resto de los sistemas.

Disponiendo de una clasificación de este tipo ya se puede responder razonadamente a preguntas como la que nos planteábamos al principio. La diferencia entre el sistema solar y la célula es que el primero es un sistema que pertenece al nivel 2, el de las máquinas, con movimiento predeterminado y que puede estudiarse con las leyes de la Física clásica. La célula, por el contrario, es un sistema de nivel 4, un sistema abierto, estructuralmente automantenido, lo que se traduce en una capacidad de supervivencia y adaptación que no tienen los sistemas de nivel 2.

Ésta es la utilidad principal de este tipo de clasificaciones, que nos permiten identificar la complejidad inherente y relevante en cada sistema como si dispusiéramos de un mapa que nos proporcionara los datos necesarios para reconocer el terreno en el que nos movemos.

3.1 Complejidad en la ciencia

Los seis primeros niveles de la jerarquía de Boulding se corresponden aproximadamente con sistemas relativos a lo que tradicionalmente se considera como "ciencia", en contraposición a "letras" o humanidades. Matemáticas, Física, Química, Biología, Zoología, Botánica, etc. Cada nivel exhibe unas determinadas características que se traducen en una complejidad propia de los sistemas que pertenecen a ese nivel. La división de la que antes hablábamos separa la complejidad que emana de factores como los considerados por Yates de la complejidad que surge con la vida (ver la clasificación de Yates, citado por Flood, en el capítulo de **Marcos Conceptuales**).

En los tres primeros niveles de la jerarquía sí es posible hablar de no linealidades, de muchos grados de libertad, de asimetrías, etc. En estos niveles se puede aplicar generalmente el principio de superposición y las interacciones están limitadas por

niveles jerárquicos, es decir, las interacciones sólo se producen entre iguales, de ahí la posibilidad de aplicar la simplificación. En el estudio de la dinámica del sistema solar se puede despreciar la existencia de partículas en el espacio y considerar únicamente las interacciones entre masas comparables. En la caída de un cuerpo suficientemente pesado se puede despreciar la fricción del aire. En una máquina de estados finitos se puede prescindir del entorno para predecir su comportamiento. Son sistemas cerrados con un comportamiento más o menos determinista. La complejidad en estos sistemas radica en nuestra falta de conocimiento sobre las leyes que rigen su comportamiento, las escalas a las que hay que operar, por ejemplo en microelectrónica, y en general la aparición de cualquiera de los factores que propone Yates.

En los tres niveles siguientes sólo se puede hablar propiamente de multiplicidad de interacciones a todos los niveles, un fenómeno que no se da en los inferiores. Aquí ya no es aplicable, salvo en raras excepciones, el principio de superposición y las simplificaciones han de hacerse con mucho más cuidado. No existen criterios tan evidentes como antes para poder hacer suposiciones sobre la importancia de un determinado elemento y grandes tendencias del comportamiento dependen a veces de una cantidad muy pequeña de una sustancia química. Los mecanismos son mucho más complejos, no sólo por nuestra falta de conocimiento sobre ellos sino por su propia estructura donde las interacciones múltiples juegan un papel decisivo en el comportamiento. Procesos como la reproducción, el crecimiento o la muerte son infinitamente más complejos que cualquier problema que se pueda plantear en los tres niveles anteriores. Esta complejidad alcanza un máximo cuando el objeto de estudio es el cerebro humano. En él se produce un salto cualitativo de los procesos meramente físicos, los seis niveles anteriores, a los procesos de orden psíquico que ya entran dentro de las categorías que distinguen los niveles 8 y 9.

Esta jerarquía de sistemas recoge el principio del que ya hablábamos al principio: el todo es más que la suma de las partes. Cada uno de estos niveles aparece cuando surge "algo más", varios átomos forman una molécula, varias moléculas un compuesto químico, determinados compuestos químicos unidos pasan a formar una cadena de ADN, frontera entre la química y la biología; un paso más sería la célula, formada por multitud de compuestos químicos pero ciertamente algo más que eso, varias células forman un tejido, varios tejidos un órgano y varios órganos un ser superior. Y aún podemos ir más lejos, un ser humano puede verse desde el punto de vista de la anatomía o de la medicina, pero también de la psicología; varias personas forman una familia o un grupo social, varias familias una comunidad, una ciudad, una región, una nación, una sociedad.

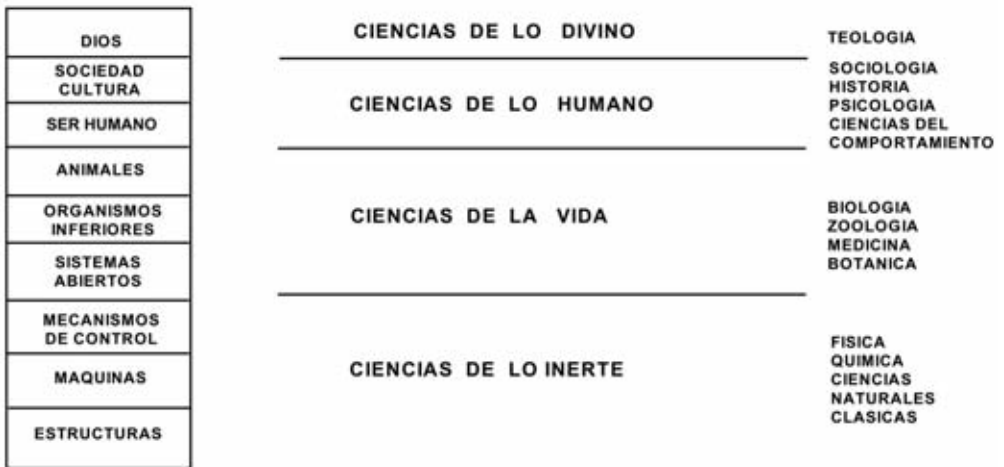


Fig. 1. Jerarquía de sistemas y las ciencias asociadas

Cada uno de estos sistemas es más complejo que el anterior y esto se refleja en las ciencias que los estudian. Checkland [Checkland, 1981] propone una clasificación de la ciencia de acuerdo con esta jerarquía de complejidad:

Física -- Química -- Biología -- Psicología -- Ciencias Sociales

Jerarquía que refleja el eje "hard-soft" de la ciencia, desde las más duras a las más blandas y, en lo que a nosotros interesa, un mapa de niveles de complejidad en la ciencia. Al igual que sucede con la jerarquía de sistemas propuesta por Boulding, con la que guarda bastante paralelismo, lo que sucede en un determinado nivel no puede explicarse sólo con conocimientos relativos a niveles inferiores -en esa escala la Física ocuparía el lugar más bajo-. Y también pone de manifiesto el mismo hecho: que las ciencias sociales son las que se enfrentan a más complejidad, en el cuadro adjunto se explica en detalle este último punto.

3.2 Complejidad en el ser humano

El humano introduce una complejidad radicalmente distinta a todas las anteriores. Conceptos como la no linealidad o la multiplicidad son casi absurdos cuando se habla de la sociedad o de cualquier fenómeno humano. La multiplicidad, la no linealidad, la no holonomicidad y la asimetría son presupuestos de partida básicos, no factores que aparezcan en determinados casos. A éstos se puede añadir la borrosidad, la desorganización, la irracionalidad o la inestabilidad permanente, características siempre presentes en todo lo humano.

Jerarquías, ciencias y lenguajes

La jerarquía de ciencias que propone Checkland es un ejemplo magnífico para enlazar estas ideas con otros conceptos tratados en estos apuntes. En virtud de la constante limitación humana para abordar la comprensión y descripción de la complejidad, cada ciencia maneja un lenguaje distinto con una precisión también distinta. Las ciencias más "hard", como puede ser la ingeniería, las matemáticas o la física trabajan con una precisión muy elevada: puentes, estructuras metálicas, máquinas, materiales, ecuaciones, son ejemplos de los objetos que se manejan en uno de los extremos del eje. Pero a medida que se avanza hacia el otro extremo la imprecisión empieza a ser mucho mayor y prueba de ello es la diferenciación entre los lenguajes que se utilizan.

Esto podemos verlo si apreciamos la similitud que tiene con el modelo jerárquico del ordenador que proponen Bell y Newell, cada nivel, en este caso cada ciencia, tiene su propio lenguaje, lenguajes que se van haciendo cada vez más borrosos y difusos a medida que subimos por esa jerarquía hacia el extremo "soft".

También se puede interpretar a la luz de la jerarquía epistemológica de sistemas que propone Klir (recogida en el capítulo de "Marcos Conceptuales"). Y, sobre todo, siguiendo sus principios generales de la complejidad, que nos indican que si reducimos la complejidad descriptiva aumenta la complejidad relacionada con la incertidumbre y viceversa. En el extremo "hard" la complejidad descriptiva es máxima (en muchos casos se llega hasta los átomos, por ejemplo en la física del estado sólido o la física de partículas) y por ello la incertidumbre muy pequeña. En el otro extremo, las ciencias blandas reducen la complejidad descriptiva al máximo y de ahí la enorme incertidumbre que se maneja en ciencias como la sociología, la psicología, la historia, ciencias del comportamiento, etc.

Para terminar este cuadro queremos resaltar una peculiaridad muy interesante. Hasta las ciencias más duras, como la física o la química, se encuentran con el dilema de no poder describir todo y tener, por tanto, que introducir la incertidumbre en sus cálculos. Esta es la misma discusión que se establece en torno al Azar (ver el capítulo dedicado a los conceptos asociados a la complejidad).

Aquí ya no existen criterios de simplificación y mucho menos casos en que el principio de superposición sea medianamente aplicable. Este hecho se puede ver muy claramente si consideramos el tipo de jerarquía distinguible en unos sistemas y en otros. En los sistemas físicos y biológicos la jerarquía se hace de acuerdo con criterios espaciales o de proximidad. Partícula subatómica, átomo, molécula y gas, o mitocondrias, membrana, núcleo, célula, tejido y órgano, son ejemplos de jerarquías en los sistemas no humanos. Estas jerarquías se distinguen al ser localizadas espacialmente subestructuras diferentes dentro de una estructura más grande. Son jerarquías de elementos. En cambio, en los sistemas humanos las jerarquías no se establecen por proximidad espacial o tamaño de los elementos que las forman, sino por interacciones. Son jerarquías de interacciones [Simon, 1969]. Sólo este hecho ya pone de manifiesto la diferencia de complejidad que va a existir entre unos sistemas y otros.

4. Sistemas sociales: una complejidad distinta

Simon no es el único autor que ha destacado la importancia de los sistemas sociales desde el punto de vista de su complejidad. Checkland propone una clasificación de sistemas, de la que no nos ocuparemos aquí de momento, en las que aparecen los Sistemas de Actividades Humanas donde se pretende recoger la emergencia de "... una dificultad que está más allá de la mera complejidad ..." [Checkland, 1981]. Flood dibuja un mapa bidimensional de las ciencias en el que el eje ciencias blandas y ciencias duras -la más dura, la física, las más blandas las ciencias sociales- se desdobra en una clasificación más avanzada al considerar la línea Homo Sapiens, es decir, la de las ciencias relacionadas con el humano -Psicología y Ciencias Sociales- [Flood, 1987]. Otros autores hablan, por ejemplo, de complejidad organizada relacional [Clemson, citado en Flood, 1987], intentando resaltar el hecho de que el ser humano introduce una complicación adicional en los sistemas y en la ciencia. Todos estos ejemplos muestran la importancia del factor humano en lo que al tratamiento de complejidad se refiere.

Aunque inicialmente pensado para aplicarlo en informática, Sáez Vacas propone también un modelo de complejidad que recoge la importancia del factor humano [Sáez Vacas, 1983] (recogido en el capítulo de **Marcos Conceptuales**). En él distingue tres niveles de complejidad, el primero de ellos es la complejidad de los elementos aislados. En informática esta sería la complejidad de los algoritmos o de los circuitos. El segundo nivel de complejidad se considera cuando ya no se trabaja con estos elementos aislados sino con sistemas, que introducen una complejidad distinta a la de los elementos. En informática ésta sería la complejidad de un ordenador completo, de un programa, de una red de ordenadores, etc. Estos dos niveles corresponden a la concepción tradicional de informática, el núcleo "duro" de la tecnología en el que trabajan los ingenieros, informáticos y tecnólogos. Pero existe aún un nivel más. Cuando esos sistemas tecnológicos se ponen en contacto con la sociedad a través de los usuarios y de los entornos de trabajo donde se implementan

aparecen problemas inabordables desde el punto de vista de la complejidad de segundo nivel, esto se debe a que ya no es una complejidad puramente técnica sino complejidad sociotécnica en la que el hombre y la sociedad juegan un papel definitivo.

5. Resumen

Comprender qué es un sistema y los diferentes tipos que existen puede ser muy útil a la hora de enfrentarse con la complejidad. En primer lugar, identificar un sistema como de una determinada clase puede orientarnos mucho a la hora de saber con qué orden de complejidad nos vamos a enfrentar y por qué. En segundo lugar, permite establecer diferencias entre sistemas con un criterio suficiente. Nuestro objetivo en este capítulo ha sido trazar una panorámica general e intuitiva sobre las distintas ciencias, sus lenguajes y la precisión que se puede esperar de ellas.

Esto último nos ha permitido distinguir la complejidad de los sistemas sociales y situarla por encima de la del resto de los sistemas, algo de primordial importancia en el campo de los ordenadores, por citar un ejemplo que nos es más cercano, pues indica que va a existir una problemática social y humana por encima de los problemas puramente tecnológicos, ya sean hardware o software.

Para ilustrar de alguna forma esta realidad podemos intentar asociar parámetros a la jerarquía de Boulding y de acuerdo con éstos relacionar a cada sistema una o varias fuentes de complejidad. No deja de ser un tanto arbitrario, sobre todo cuando se trata de sistemas superiores y más teniendo en cuenta que no disponemos de ningún criterio para hablar de fuentes de complejidad en sistemas como pueden ser los animales, el hombre o la sociedad. Aún así, nosotros proponemos las siguientes correspondencias entre sistemas, parámetros que los caracterizan y fuentes de complejidad. Y dado que la intención es más aclarar ideas que teorizar sobre la complejidad, animamos al lector a criticar ésta y proponer la suya.

SISTEMA	PARAMETROS CARACTERISTICAS	FUENTES DE COMPLEJIDAD
DIOS	?	CONOCIMIENTO INAPREHENSIBLE
SOCIEDAD CULTURA	IDEALES	IRRACIONALIDAD
SER HUMANO	META CONOCIMIENTO	INDETERMINISMO
ANIMALES	CONOCIMIENTO	OBJETIVOS DINAMICOS
ORGANISMOS INFERIORES	ORGANIZACION	SUPERSISTEMAS ESPECIALIZADOS
SISTEMAS ABIERTOS	ADAPTACION	INTERACCIONES
MECANISMOS DE CONTROL	INFORMACION	NO HOLONOMICIDAD
MAQUINAS	TIEMPO	NO LINEALIDAD
ESTRUCTURAS	MASA TAMAÑO	ASIMETRIAS MULTIPLICIDAD

Fig. 2. Sistemas, características y fuentes de complejidad

6. Bibliografía

La dividimos en dos apartados. En el apartado de Notas Bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

Muchas de las referencias utilizadas en este capítulo han aparecido ya en otros anteriores por lo que no nos vamos a detener en ellas, sin embargo, sí queremos destacar el libro **Systems Thinking, Systems Practice**, de Peter Checkland [J. Wiley & Sons, Chichester, 1981], en el que desarrolla la teoría y metodología de los sistemas blandos. La primera parte del libro es un repaso, ameno e interesante, de la historia de la ciencia desde la perspectiva de los sistemas. En la segunda parte se introduce la metodología de sistemas blandos, de una forma clara y fácil de entender. Esta metodología se puede considerar una meta-metodología por su carácter general y la presentación de la misma como herramienta de resolución de problemas. Puede incluso verse como un auténtico software intelectual para la gestión, no sólo tecnológica sino de cualquier problema de carácter sistémico y, muy especialmente, por supuesto, de aquéllos en los que aparezca el factor humano. Para los entusiastas de los sistemas ésta es una referencia imprescindible.

Referencias bibliográficas

Checkland, P. (1981) **Systems Thinking, Systems Practice**, J. Wiley & Sons, Chichester, 1981.

Flood, R.L. (1987) "Complexity: a definition by construction of a conceptual framework", **Systems Research**, vol. 4, num. 3, pp. 177-185.

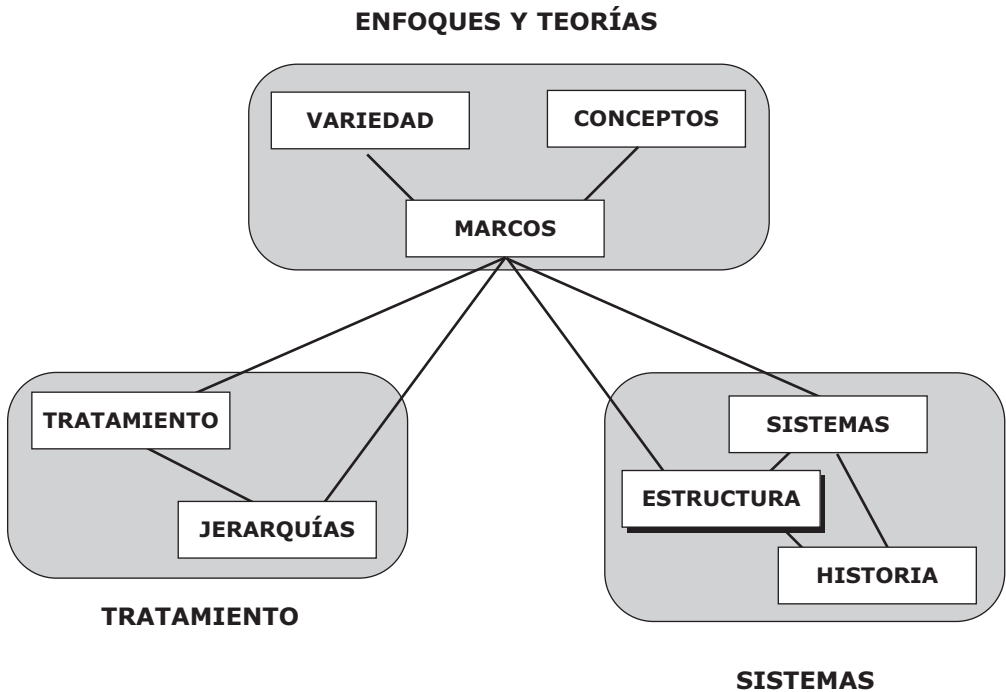
Pagels, H.R. (1989) **The dreams of reason: the computer and the rise of the sciences of complexity**, Bantam Books, N.Y., 1989.

Simon, H.A. (1969) **The sciences of the artificial**, MIT Press, Cambridge, Ma. 1969.

Capítulo 5: Sistemas: visión estructural y visión funcional

"Tanto en la ciencia como en la ingeniería, el estudio de "sistemas" está convirtiéndose en una actividad cada vez más popular. Su popularidad responde más a una acuciante necesidad de sintetizar y analizar la complejidad que al desarrollo de un cuerpo de conocimiento y unas técnicas para tratar la complejidad. Para que esta popularidad sea algo más que un capricho, la necesidad ha de dar paso a alguna invención y proporcionar una base a las ideas sobre sistemas." [Simon, 1969]

En virtud de la popularidad que reconoce Simon, la palabra sistema se ha convertido en un comodín en muchas ciencias: sistemas informáticos, ecosistemas, sistemas orgánicos, sistema de ecuaciones, sistema circulatorio, sistema de impuestos, sistema inmunológico, sistema estelar. Todos ellos no son más que unos pocos ejemplos de lo que se ha venido en llamar la "ciencia de los sistemas", es decir, el estudio de las propiedades comunes a todos los sistemas.



1. Introducción

En este capítulo vamos a presentar algunas de las ideas relacionadas con los sistemas. A lo largo de todos estos apuntes, ha aparecido repetidas veces el concepto de sistema, del que existe una noción intuitiva que se utiliza con profusión en muy diversas ciencias. El primer objetivo que nos vamos a marcar en este capítulo es definir qué es un sistema, tarea nada fácil, como veremos. Una vez definido el sistema podemos pasar a estudiar sus características, esas características generales que aparecen en todos los sistemas, independientemente de la aplicación. Los conceptos que estudiaremos son importantes desde el punto de vista de la complejidad, porque a nadie que haya leído con detenimiento estos apuntes se le escapará la estrecha relación que tienen con muchas ideas tratadas en otros capítulos. Este pretendemos que sea una pequeña guía para el estudio de los sistemas, intentando siempre basarnos en ideas intuitivas y dejar el formalismo matemático de lado.

2. Sistemas abiertos y complejidad

La complejidad en los sistemas alcanza su verdadera dimensión cuando se consideran sistemas abiertos, es decir, sistemas que interactúan con su entorno, viéndose influidos por las variaciones de éste e influyendo a su vez en él. Puede parecer trivial considerar sistemas abiertos y sistemas cerrados pues muy pocos sistemas, excepto los artificiales, se pueden considerar como cerrados. La diferencia está en la complejidad.

Durante mucho tiempo, en diversos campos de la ciencia se estudiaban única y exclusivamente sistemas cerrados, considerar la más mínima interacción con el exterior suponía un aumento inabordable de la complejidad. La economía es un ejemplo de ello, sólo recientemente se ha empezado a ver la importancia del entorno en muchos aspectos: el entorno ecológico, las comunicaciones, la existencia de un mercado mundial, las repercusiones sociales de determinadas políticas, etc. La biología, como en muchas otras cosas, fue una de las primeras ciencias en ver la necesidad de ampliar los estudios de los objetos al entorno que los circunda. El ejemplo más claro de ello es la teoría de la evolución. Pero en la mayoría de los casos, considerar sistemas abiertos es más fácil de decir que hacer. En electrónica, por ejemplo, hasta el más novato de los estudiantes sabe que hay multitud de interferencias, ruidos, y efectos parásitos que provienen del entorno del circuito, sin embargo, la teoría de circuitos procede como si estos fueran sistemas perfectamente aislados y la protección contra el ruido es aumentar la potencia de la señal para que supere a éste. ¿Es factible desarrollar una teoría de circuitos que tenga en cuenta todas las interacciones con el entorno?, y esas interacciones no se limitan a las interferencias, hay intercambio de calor, influencia de la humedad en el ambiente, temperatura de trabajo, resistencia física de los materiales e infinidad de otros parámetros en los que la mayoría de las veces ni siquiera se piensa.

Y no es exagerado hablar de sistemas "tan abiertos". En muchos ordenadores actuales, la arquitectura hardware está preparada especialmente para poder soportar esos errores "inexplicables" que aparecen por el simple hecho de que el ordenador está en un entorno real (rayos cósmicos, variaciones de las condiciones de trabajo, aleatoriedad cuántica, etc.).

3. Definición de sistema

Es difícil definir la noción de sistema. Klir [Orchard, 1977] reconoce que existe una teoría de sistemas para cada definición que se adopte y que cada definición está condicionada por el problema que se intenta resolver. Existen multitud de definiciones que avalan esta afirmación. De hecho ésta es una de las críticas que se hacen más frecuentemente al enfoque sistémico como teoría general, ya que muchas de las definiciones que se utilizan dependen en gran medida de la aplicación.

Existen, sin embargo, una serie de rasgos genéricos que recogen la mayoría de las definiciones de sistema, entre ellas, aunque no necesariamente por ese orden, la existencia de varios elementos que se relacionan entre sí y la existencia de un objetivo común. La interpretación de estas características comunes, así como el hecho de resaltar unas más que otras, es lo que generalmente viene condicionado por la aplicación que se quiere hacer de la definición. Por citar alguna, a continuación se recoge la definición de sistema que propone Gómez-Pallete, definición que utiliza en un estudio sobre organización y gestión de empresas:

Un sistema es

1. Un conjunto de elementos
2. Relacionados entre sí
3. Actuando en un determinado entorno
4. Con el fin de alcanzar objetivos comunes
5. Y con capacidad de autocontrol

Klir sostiene que existen diferentes posibles definiciones de sistemas, según cómo se interpreten estos últimos. De acuerdo con su jerarquía epistemológica de sistemas (ver el capítulo de **Marcos Conceptuales**), y su estudio formal de los mismos, propone una serie de definiciones que recogemos en el cuadro adjunto.

Como definición básica de los posteriores trabajos de Klir, podemos citar la definición de sistema de Ashby [1956]: "Cuando se establece un conjunto de variables como resultado de nuestra interacción con el objeto que nos interesa, entonces se dice que se distingue un sistema en el objeto".

Las cinco definiciones básicas que propone Klir se basan en los rasgos fundamentales de los sistemas: cantidades observadas a un nivel determinado de resolución, actividad de las cantidades en el tiempo, relaciones invariantes en el tiempo entre esas cantidades, y propiedades que determinan estos rasgos. A partir de estas cinco definiciones básicas se pueden establecer otras definiciones en función de rasgos permanentes pero no fundamentales que exhiba cada grupo de sistemas particular. Las cinco definiciones básicas son las siguientes:

Un sistema es un conjunto dado de cantidades, consideradas a un nivel de resolución dado.

Un sistema es un conjunto de variaciones en el tiempo de las cantidades que se consideran.

Un sistema es una relación dada, invariante en el tiempo, entre valores instantáneos y/o pasados y/o futuros de las cantidades externas. La relación puede admitir una interpretación probabilística pero no es necesario hacerla.

Un sistema es un conjunto dado de elementos, sus comportamientos permanentes, y un conjunto de acoplamientos entre los elementos y entre los elementos y el entorno.

Un sistema es un conjunto de estados y un conjunto de transiciones entre los estados. Se puede admitir una interpretación probabilística de la ocurrencia de transiciones de un estado a otro, pero no es necesario hacerla.

Cada una de las definiciones está incluida en la siguiente, apareciendo así una jerarquía interpretativa de los sistemas. La primera definición es el "esqueleto" del sistema, de qué elementos se va a construir. La segunda definición se apoya en la primera pero destaca la actividad del sistema, es decir, los valores que van tomando a lo largo del tiempo las cantidades que conforman el sistema. La tercera definición supone las dos anteriores y se centra en las relaciones que ligan las diferentes cantidades del sistema, que se conocen por comportamiento del sistema. La cuarta definición toma los elementos y sus comportamientos (y por tanto las tres definiciones anteriores) y define el sistema a través de éstos y los acoplamientos entre ellos. La quinta definición es una especie de visión dinámica de la cuarta, en virtud de los elementos y sus estados, el sistema se define como transiciones entre esos estados.

4. Interpretación funcional y estructural de los sistemas

Consideremos un sistema cualquiera. De alguna forma el observador ha debido definirlo, delimitando entre otros aspectos, unas "fronteras" entre lo que consideramos objeto de nuestro estudio y el entorno en el que está inmerso. A través de estas fronteras se relacionarán uno con el otro de alguna manera. Pues bien, si atendemos a estas interrelaciones para caracterizar el objeto estudiado, sin importarnos su naturaleza "interna", estamos centrándonos en el aspecto funcional. Puede resumirse esta actitud diciendo que lo que le interesa es el "qué hace". Pero también podemos mirar dentro de esa "frontera" definida, e investigar sobre los elementos que componen el sistema, las relaciones que hay entre ellos, cómo están organizados. Este tipo de enfoque es el estructural: lo importante es el "cómo lo hace".

Sea un sistema (S) como el de la figura 1, inmerso en un entorno (E). Agrupemos todas las influencias que reciba, llamando entrada (e) a este agregado. Y hagamos lo mismo con todas las formas en que el sistema actúa sobre el entorno (salidas, s).

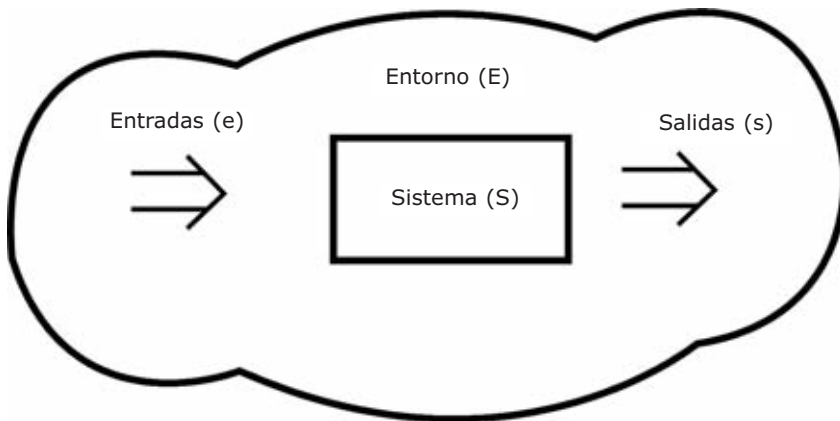


Fig. 1. Esquema correspondiente a un sistema visto desde un punto de vista funcional

Podemos pensar en el sistema como "caja negra", de la que nada sabemos sobre cómo es por dentro, pero que estudiamos en su relación con el exterior. Así buscamos una relación funcional entre la salida y la entrada, de la forma $s = S(e)$ (esta relación es la tercera definición de sistema en la jerarquía de Klir estudiada en **Marcos Conceptuales**). En este caso estamos utilizando un enfoque funcional, en el que no interesa cómo se consigue esa función de transferencia sino sólo lo que hace.

Pero también podemos considerar un sistema como un conjunto de componentes (C) e interacciones entre ellos (I). Es el caso de la segunda figura. En este caso nuestro interés se centrará en conocer por completo la estructura $S = I + C$ (que coincide con la cuarta definición de sistema de Klir).

Ahora lo que se utiliza es una aproximación estructural, en la que lo que interesa conocer es la organización interna del sistema.

Si conocemos completamente la estructura de un sistema (todos sus elementos componentes, el comportamiento de los elementos y la forma como se relacionan entre sí), tenemos los datos suficientes como para calcular la función que realiza. Sin embargo, el caso contrario no es cierto: a partir de la función que caracteriza a un objeto en su interacción con el entorno, no nos es posible definir su estructura. De aquí se deduce que la descripción estructural aporta más información (y caracteriza más completamente) que la funcional. Un ejemplo de ello se puede encontrar en la electrónica, donde es relativamente sencillo (y desde luego posible) calcular la función de transferencia de un cuadripolo si conocemos los componentes que contiene, y cómo están conectados unos con otros. Pero el caso contrario, esto es, descubrir la estructura interna del cuadripolo a partir de su función de transferencia, es imposible: hay infinitas combinaciones de elementos electrónicos que dan el mismo comportamiento de "caja negra". Para poder decidir cuál de todas ellas es la que se corresponde con una función dada, necesitamos saber a priori algo sobre la estructura interna del cuadripolo.

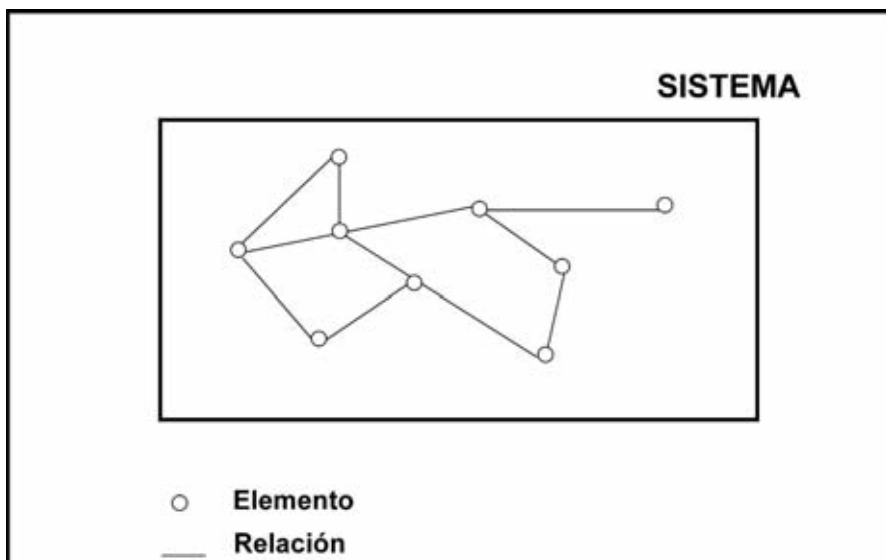


Fig. 2. Esquema de un sistema visto desde el punto de vista estructural

En informática es fácil observar lo mismo. Dado el código completo de un programa (su estructura), podemos predecir acertadamente su comportamiento (su descripción funcional). Pero un mismo comportamiento de cara al usuario puede esconder programas muy diferentes.

5. El sistema desde el punto de vista estructural

La aproximación estructural a los sistemas intenta descubrir su organización interna, qué elementos lo componen, cómo se interrelacionan, qué dependencias hay entre ellos, qué estructuras se pueden distinguir dentro de un mismo sistema, etc. Es evidente que cada sistema tiene sus propias características que lo hacen único, pero hay una serie de parámetros, por llamarlos de alguna forma, que son comunes a todos los sistemas y que pueden ser muy útiles a la hora de estudiarlos.

5.1 Límites de un sistema

Todo sistema, sea abierto o cerrado, tiene unos límites que lo definen y permiten identificarlo y diferenciarlo de otros sistemas. En los sistemas cerrados estos límites aparecen muy claramente pero en los sistemas abiertos estos límites son generalmente muy difusos y difíciles de establecer. Aún en este último caso, para estudiar un sistema se empieza por delimitarlo, separando lo que es el sistema y lo que va a ser su entorno.

Un ejemplo muy útil son las ciudades. Una ciudad se puede estudiar como un sistema abierto pero ¿cómo se establecen los límites?. Una forma de hacerlo puede ser atenerse a las divisiones municipales, otra establecer una zona de influencia de tantos kilómetros. Cualquier decisión que se tome implica, por ejemplo, que no se van a considerar las ciudades dormitorio que rodean a la ciudad que nos interesa (a pesar de que son un producto directo de ella e incluso una parte de la misma), o que las zonas rurales que rodeen a la ciudad se considerarán como entorno (cuando, como sucede a menudo, son también zonas residenciales de gente que trabaja en esa ciudad), ¿hasta dónde llega la zona de influencia de una gran capital?, ¿hasta qué punto se puede diferenciar entre entorno y zonas que están en los límites de la ciudad?.

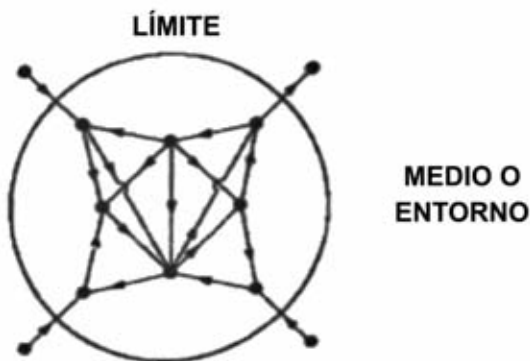


Fig. 3. Límites de un sistema [Aracil, 1986, p. 44]

Establecer los límites del sistema es una situación que se presenta muy a menudo en la ciencia y en la ingeniería, cuando se habla de centrar, acotar, definir o clarificar un problema se está intentando definir cuáles son sus límites.

5.2 Elementos

Una vez establecidos los límites del sistema ya se pueden identificar los elementos que lo componen. De ahí la importancia de seleccionar unos límites adecuados, sin una definición apropiada será imposible identificar los componentes del sistema. El ejemplo de la ciudad es muy claro, una vez delimitada sólo se considerarán elementos de ella lo que esté dentro de esos límites, lo demás será parte de su entorno. Por otro lado, al nombrar el sistema, es decir, identificar sus características relevantes que nos interesan, restringimos los elementos que se van a considerar.

Si estudiamos la ciudad desde el punto de vista medioambiental los elementos que nos interesan son focos de polución, zonas verdes, vertederos, depuradoras, etc., y hay que establecer unos límites definidos (qué tramo de un río se considera elemento de la ciudad, influencia de montañas que están cerca, fenómenos como la lluvia ácida, que puede originarse a muchos kilómetros de la ciudad que se estudia, etc.). Si lo que nos interesa es la ciudad desde el punto de vista organizativo, los elementos relevantes serán las juntas de vecinos y de distrito, los barrios, las concejalías, el ayuntamiento, etc., en cuanto a los límites hay que estudiar cómo se considera el gobierno civil, el gobierno autonómico, y todas las estructuras organizativas que no son exclusivas de la ciudad.

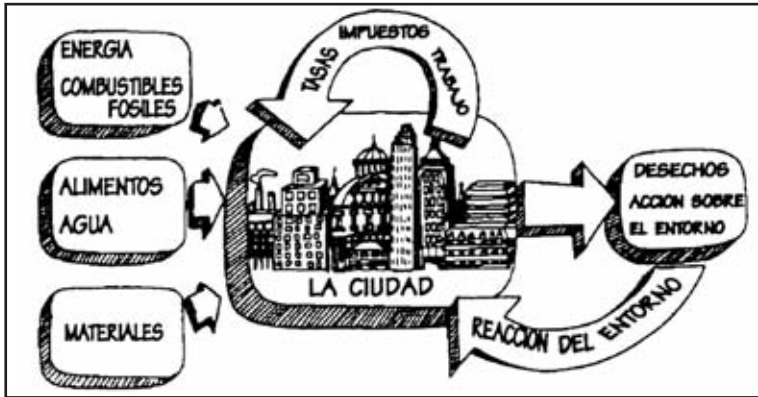


Fig. 4. Elementos que conforman un sistema [Rosnay, 1977, p. 41]

La identificación de los elementos es, como se ve, función de cómo definimos el sistema y los límites que le imponemos. Hay que tener en cuenta, además, el nivel de resolución con que queremos trabajar, no es lo mismo estudiar la organización limitándonos a las estructuras municipales que llegar hasta las juntas de vecinos. El nivel de resolución elegido nos va a indicar hasta donde tenemos que considerar el sistema. Pueden existir varios niveles de resolución, en cuyo caso, los elementos se transforman a su vez en subsistemas sobre los que de nuevo se definen unos límites, unos componentes y un nivel de resolución. La ciudad desde el punto de vista organizativo presenta muchos niveles de resolución, la familia, los grupos de vecinos, los distritos o los barrios, y muchos subsistemas como la escuela (organizan la educación), las oficinas y las fábricas (organizan el trabajo), los hospitales (organizan la sanidad), la policía (organizan la seguridad), etc.

5.3 Interacciones

La condición para que los elementos considerados formen un sistema es que interactúen entre ellos con un objetivo común (ver las definiciones de sistemas). Las interacciones no tienen por qué ser permanentes, pueden producirse sólo en momentos determinados, sólo ocasionalmente, de forma aleatoria, y tampoco tienen por qué ser lineales ni separables. Una gran parte de la complejidad de los sistemas proviene de las interacciones entre sus elementos.

Las interacciones se pueden dar entre niveles muy diferentes y entre todos los subsistemas, por ello es muy importante elegir bien el nivel de resolución y cómo se agrupan los elementos para formar subsistemas. Los gases se pueden estudiar evitando mucha de su complejidad disminuyendo el nivel de resolución, interesándonos sólo por propiedades del todo (presión, volumen, temperatura), si aumentamos el nivel de resolución nos encontramos con un número enorme de interacciones entre moléculas que nos impide hacer cualquier previsión formal

sobre el comportamiento del gas. En una ciudad no es buena idea agrupar a los habitantes por orden alfabético, pues este agrupamiento no tiene nada que ver con las interacciones entre ellos, los grupos que se hacen se definen en función de unas interacciones comunes: la familia, los universitarios, los menores de 18 años, los estudiantes de ingeniería, los habitantes de un barrio, los socios de un determinado club, etc.

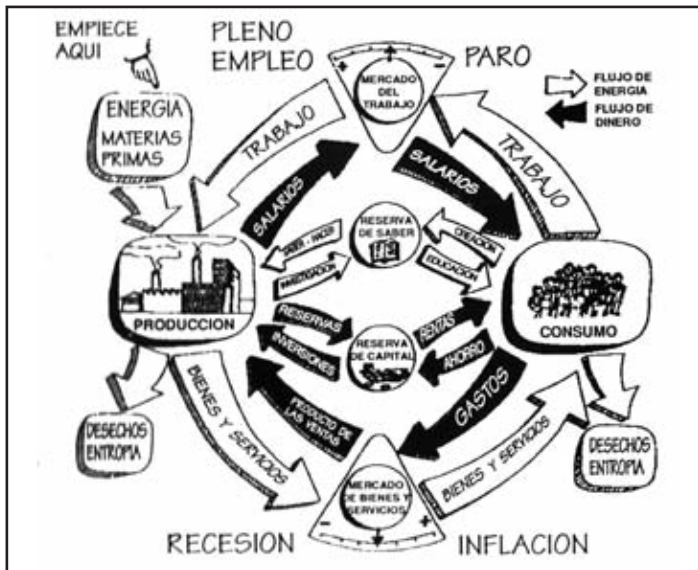


Fig. 5. Interacciones que componen un sistema [Rosnay, 1977, p. 28]

Una correcta elección de los elementos que se consideran en el sistema, junto con el nivel de resolución más adecuado puede ayudar mucho a reducir la complejidad derivada de las interacciones entre los elementos.

5.4 Entorno

Los sistemas abiertos no quedan completamente caracterizados si no se tiene en cuenta el entorno. De forma intuitiva, el entorno de un sistema es todo aquello que lo rodea y condiciona en su comportamiento y evolución. Existe una relación bidireccional entre el sistema y el entorno dado que interactúan entre ellos, el sistema modificando el entorno y el entorno, al obligar al sistema a adaptarse a él.

La importancia del entorno es considerable en todos los sistemas, pero es fundamental para poder entender el funcionamiento de sistemas evolutivos, autoorganizados y adaptativos. El entorno de una ciudad es muy amplio: el área física en la que está, la región, la provincia, la nación, etc., y no se la puede estudiar a fondo como sistema si no se considera debidamente el entorno en el que ésta está situada.

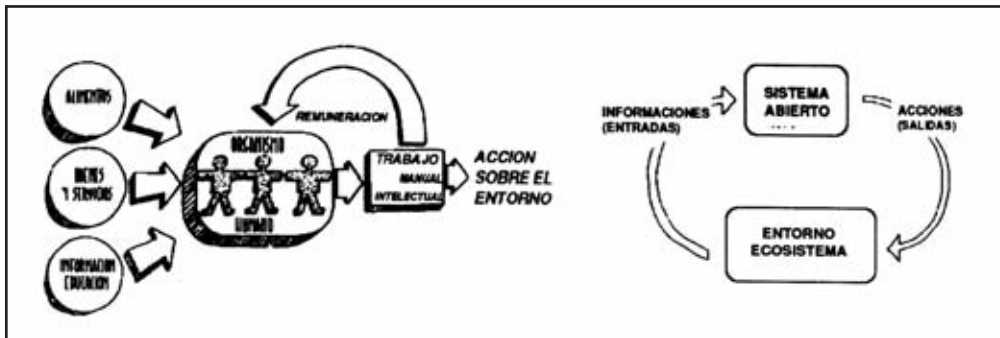


Fig. 6. Entorno de diferentes sistemas [Rosnay, 1977, pp. 52 y 82]

Esto nos lleva a un punto importante respecto al entorno. Al igual que trabajábamos con un nivel de resolución al definir los elementos, hay que limitar la influencia del entorno a un nivel relevante, si estudiamos la ciudad organizativamente quedándonos en la administración municipal, no podemos considerar como entorno los grupos familiares del extrarradio. Por la misma razón, al considerar subsistemas, hay que identificar el entorno relevante de cada uno de ellos, entorno con el que interactúa y que será parte del entorno total del sistema.

5.5 Jerarquía de niveles

Este aspecto es fundamental dentro del estudio de los sistemas y, en general, en el estudio de la complejidad (ver el capítulo dedicado a las **Jerarquías**). Ya hemos visto cómo los elementos se pueden agrupar en subsistemas, que a su vez pueden agruparse entre sí formando otros subsistemas. Se forma así una jerarquía de niveles en el sistema, cada uno de los cuales se puede estudiar como un sistema en sí mismo.

Cada nivel de la jerarquía tiene un nivel de resolución, se relaciona con un determinado entorno y se caracteriza por un tipo determinado de interacciones. Generalmente, cada nivel de una jerarquía lleva asociado, además, un lenguaje propio que le caracteriza y no se puede entender en otros niveles. El ordenador, tal y como se muestra en la figura 7, es un ejemplo muy claro de jerarquía: puede

verse como un sistema genérico, pero dentro de él podemos distinguir diversos niveles con subsistemas característicos (ver como referencia el capítulo dedicado al **Tratamiento de la complejidad** donde se estudia a fondo este tema). El nivel más bajo puede ser el de circuito, después el de puertas lógicas, por encima de éste el nivel de registros, el de microprogramas, el lenguaje máquina, los lenguajes estructurados, los programas, etc. En cada uno de estos niveles hay distintos sistemas: circuitos electrónicos, circuitos lógicos, secuenciadores, unidades de proceso, etc., cada uno con unos límites y un entorno definidos y cada uno de ellos con un lenguaje distintivo: la teoría de circuitos, el álgebra de Boole, la teoría de autómatas, el ensamblador, lenguajes de programación avanzados, interfaces, etc.

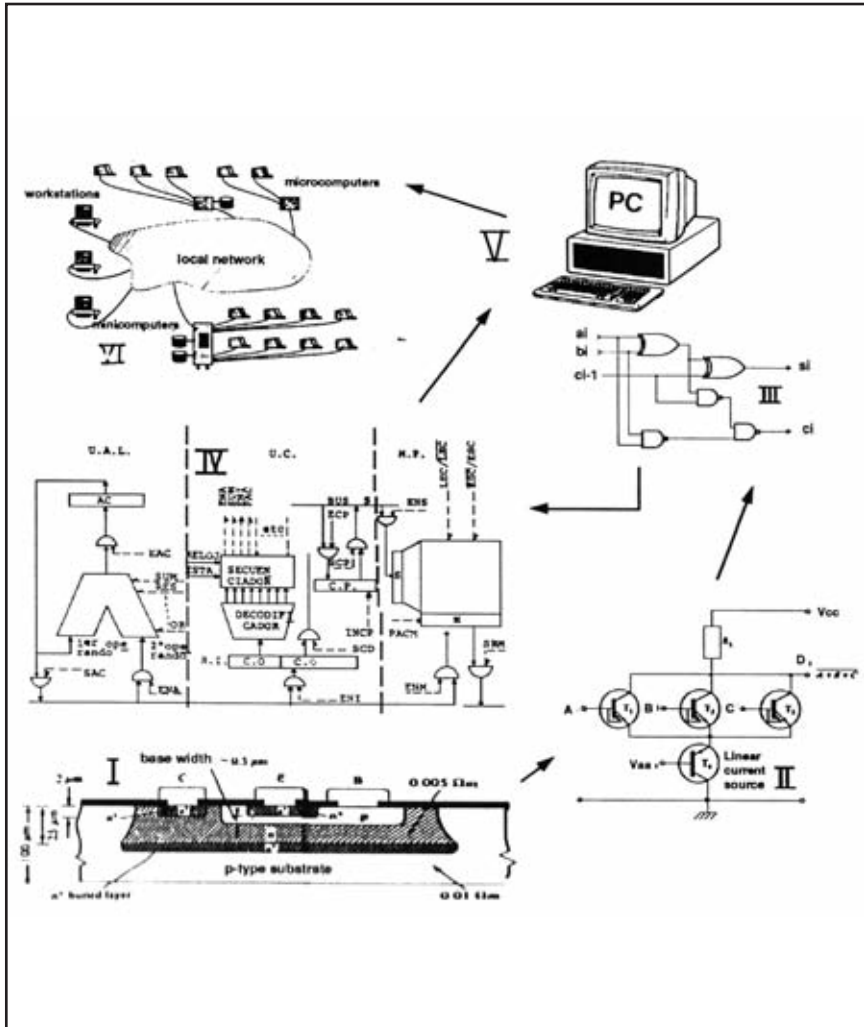


Fig. 7. Jerarquía de niveles de un posible sistema

Las jerarquías ayudan a reducir la complejidad en el estudio de un sistema al permitir separar de una forma ordenada diferentes partes del mismo y situarlas en unos niveles interdependientes. Al mismo tiempo permiten trabajar en diferentes niveles de resolución, aislar partes del sistema global para estudiarlas más a fondo sin entrar en la complejidad del resto del sistema.

6. El sistema desde el punto de vista funcional

La aproximación funcional a los sistemas intenta describir su funcionamiento con un planteamiento de "caja negra", es decir, sin atender a las estructuras, interacciones y elementos que hacen posible ese funcionamiento. La mejor caracterización de este tipo de aproximación es la que antes veíamos: se trata de estudiar la relación entre las entradas al sistema y las salidas que produce. Se trata de ver qué hace "el sistema".

Al igual que sucede con la aproximación estructural, dentro de la visión funcional hay una serie de conceptos genéricos que sirven para caracterizar al sistema y facilitar su estudio. Muchos de los conceptos que veremos a continuación están muy relacionados con otros capítulos en donde se estudian aspectos más detallados, es conveniente no perder de vista estas referencias.

6.1 Entrada y Salida

Lógicamente, para caracterizar un sistema funcionalmente lo primero que hay que definir es qué es la entrada al sistema y qué la salida. La entrada será cualquier flujo, sea material o de información, que vaya del entorno al sistema, la salida será cualquier reacción del sistema, material o de información, que vaya hacia el entorno. La diferencia que se hace entre flujos materiales y de información es importante pues no siempre se puede evaluar la entrada en términos cuantitativos.

En un circuito electrónico la entrada puede ser una cierta tensión, la salida la corriente que circula por un amperímetro. Pero también puede que la entrada sea una temperatura si el circuito es sensible a este parámetro y la salida no es sólo corriente o tensión sino también calor disipado hacia el exterior. Generalmente no se consideran las entradas o salidas que no son suficientemente grandes o no parecen relevantes, pero no por ello dejen de estar presentes, tómesese como ejemplo las interferencias, una "entrada" al circuito que está siempre presente y puede ser suficientemente importante como para distorsionar la salida. Es importante tener en cuenta estas entradas adicionales a los sistemas pues pueden llegar a ser decisivas en el comportamiento global (ejemplo de ello es el envejecimiento de los componentes que convierte un circuito determinista en caótico, tal y como veíamos en el capítulo de conceptos relacionados con la complejidad).

6.2 Realimentación, positiva y negativa

Uno de los mecanismos básicos del funcionamiento de los sistemas es la realimentación. La cibernética, por ejemplo, basa gran parte de sus estudios en ella. Básicamente consiste en que la salida del sistema se convierte, de alguna manera, en una entrada más del sistema. De esta forma el sistema puede "saber" qué es lo que está haciendo y modificar su comportamiento en consecuencia. Hay muchos ejemplos de realimentación, uno de los más sencillos es el de los servomecanismos simples que definen la próxima salida en función de la entrada y de la salida anterior. El ejemplo clásico en la cibernética son los mecanismos de tiro automático que calculan la nueva posición del cañón en función del objetivo y la posición actual.

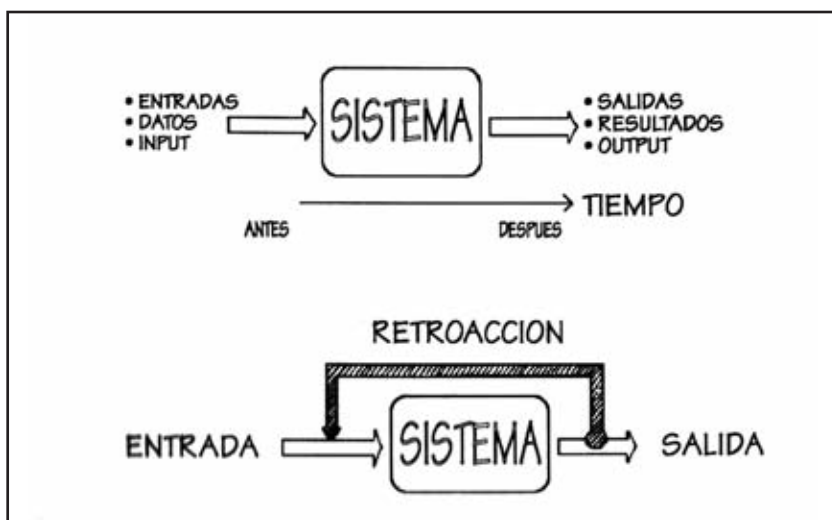


Fig. 8. La realimentación como elemento funcional de los sistemas [Rosnay, 1977, p. 89]

Por lo general, la realimentación se hace en forma de algún tipo de información de la salida hacia la entrada y según sea el efecto que produce esta información así será la realimentación. Si la realimentación produce resultados del mismo tipo que los resultados precedentes se trata de una realimentación positiva. Si, por el contrario, los resultados son de signo contrario a los resultados anteriores se trata de una realimentación negativa. Ejemplos de realimentación positiva son la relación entre la población y el número de nacimientos (más población ==> más nacimientos ==> más población ...), o la fisión nuclear (más fisión ==> más partículas ==> más colisiones ==> más fisión ...). Ejemplos de realimentación negativa son la población y la tasa de natalidad (más población ==> más mortalidad ==> menos población ...), o los termostatos (menos temperatura ==> más calefacción ==> más temperatura ...).

La realimentación positiva tiene efectos acumulativos y conduce a inestabilidades o a bloqueos. Cuando existe una realimentación positiva un aumento de la salida provoca que la salida siga aumentando indefinidamente, si se produce una disminución, la salida disminuirá continuamente hasta hacerse nula. El ejemplo de la población ilustra muy bien este punto.



Fig. 9. Ejemplo de realimentación positiva [Rosnay, 1977, p. 90]

La realimentación negativa produce una convergencia hacia un valor determinado, por ello se aparece en los sistemas que tienen un objetivo. El comportamiento aparentemente inteligente de los dispositivos con realimentación negativa fue lo que llamó la atención a los primeros estudiosos de la cibernética (ver el capítulo dedicado a la **Historia del Enfoque Sistémico**).

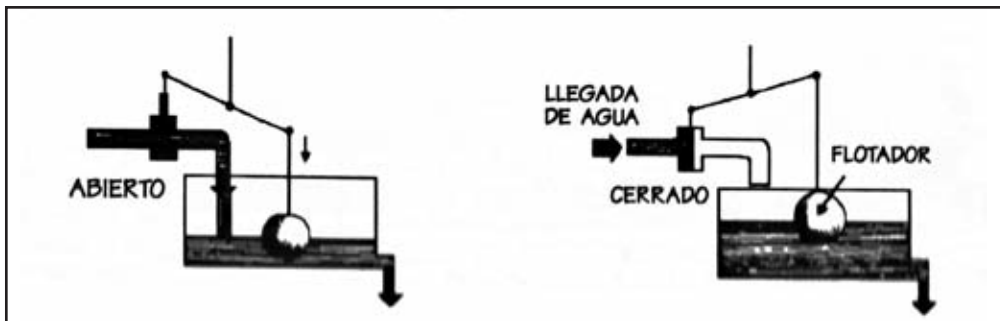


Fig.10. Ejemplo de realimentación negativa [Rosnay, 1977, p. 91]

En un sistema suelen aparecer varios bucles de realimentación, positiva y negativa, generalmente interrelacionados entre sí lo que a veces dificulta identificarlos como tales. Es muy común encontrar un bucle negativo asociado a uno positivo.

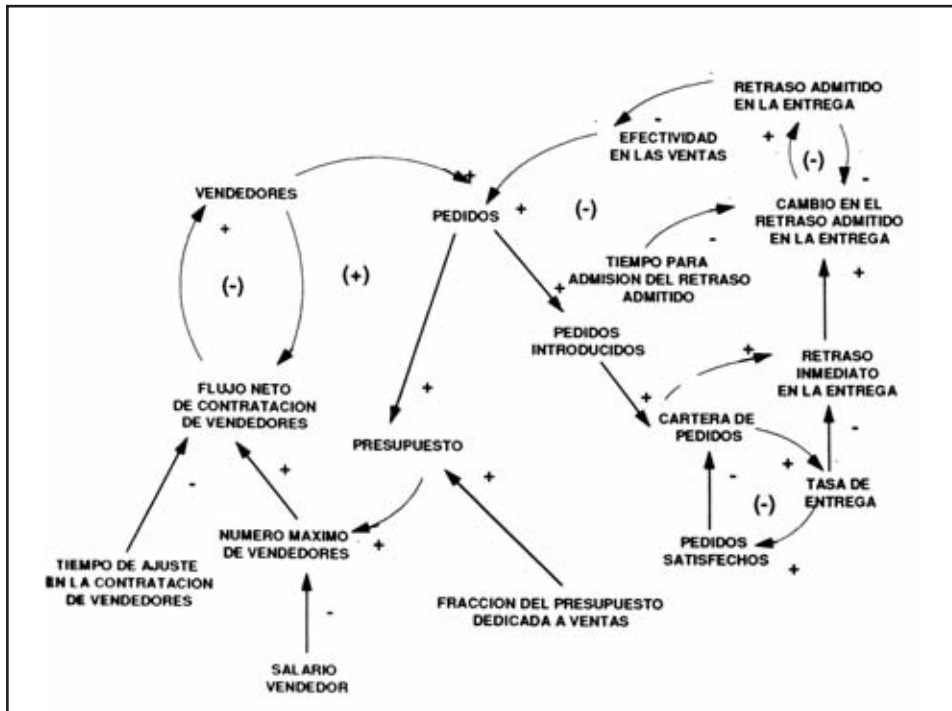


Fig. 11. Sistema con varios bucles de realimentación [Aracil, 1986, p. 124]

6.3 No linealidades

Ya hemos visto en capítulos anteriores qué son las no linealidades y su influencia en la complejidad de los sistemas (ver "Marcos Conceptuales" y el apartado dedicado al caos en "Conceptos asociados a la complejidad"). Desde el punto de vista funcional, la presencia de no linealidades es decisiva pues significa que existen puntos de bifurcación donde el comportamiento del sistema cambia cualitativamente.

En un sistema lineal, un aumento de la entrada conduce a un aumento de la salida, en un sistema no lineal un aumento en la entrada puede conducir a un estado totalmente nuevo. Las no linealidades pueden aparecer en un momento cualquiera de la vida del sistema, como sucede en los circuitos electrónicos por envejecimiento de los componentes, y también pueden estar presentes desde el origen del sistema pero no aparecer sino sólo bajo ciertas condiciones que rara vez se dan. Por ello es difícil predecir el comportamiento de sistemas donde esto puede suceder.

El estudio de las no linealidades es difícil y aún lo es más tratar con ellas. La ciencia del caos ha proporcionado algunas ideas más sobre su naturaleza y efectos pero todavía faltan herramientas que permitan entenderlas y manejarlas mejor.

6.4 Equilibrio y estabilidad

La noción que tenemos de estabilidad suele estar cerca de lo que se entiende por sistema estático. Sin embargo, un sistema estable es aquél que mantiene sus características e identidad a lo largo del tiempo. Los sistemas en equilibrio pueden reaccionar de dos formas ante perturbaciones externas: en los sistemas en los que predomina la realimentación negativa y no hay linealidades, una perturbación alejará el sistema del equilibrio pero éste volverá a él tras un tiempo (conocido como tiempo de relajación), si en el sistema hay no linealidades o predomina la realimentación positiva, las perturbaciones lo alejarán del equilibrio y el propio sistema se alejará aún más de él para entrar en un nuevo modo de equilibrio o desaparecer.

Para que un sistema con no linealidades cambie su comportamiento, las perturbaciones, o la entrada, han de superar un cierto umbral y que se alcance un punto de bifurcación (llamado así porque en él se produce un cambio cualitativo en el comportamiento del sistema). En un sistema donde predomine la realimentación positiva basta la más mínima alteración del punto de equilibrio para que el sistema no vuelva a entrar en equilibrio.

En los sistemas estables, el concepto de estabilidad también es relativo pues está en función del tiempo de relajación que antes mencionábamos. Si las perturbaciones llegan al sistema más rápido de lo que éste puede asimilarlas se entrará en una situación permanente de inestabilidad.

6.5 Adaptación y cambio

Después de lo que decíamos en el apartado anterior se hace difícil pensar en algún sistema que sea estable. Y, sin embargo, existen muchos sistemas que sobreviven a las perturbaciones, la prueba más evidente de ello es que existen. Una célula, un animal, el hombre o las empresas son ejemplos de sistemas supervivientes. Esto no quiere decir que sean inmunes a las perturbaciones, sino que han podido adaptarse a ellas.

La biología nos ofrece ejemplos magníficos de sistemas que se han adaptado a entornos cambiantes (perturbaciones) y han sobrevivido. La evolución de los seres vivos es una cadena de adaptaciones, de cambios en la estructura del sistema para poder adaptar su funcionalidad al nuevo entorno. Esta peculiaridad es una de las más difíciles de reproducir en los sistemas artificiales.

La viabilidad de cualquier sistema pasa por su capacidad de adaptación y cambio. Los sistemas naturales poseen esta importante cualidad y el estudio de los mecanismos de adaptación es uno de los campos más importantes dentro de los sistemas construidos por el hombre.

6.6 Evolución y emergencia

Directamente relacionados con los dos conceptos anteriores, la evolución y la emergencia son dos de las características fundamentales de la funcionalidad de un sistema. El hecho de que un sistema pueda adaptarse al entorno le permite superar ciertas "agresiones", si además puede cambiar para hacer frente a las nuevas situaciones se dice que el sistema evoluciona. Al igual que antes ésta es una característica normal en los sistemas naturales, pero pocas veces se puede hablar de un sistema artificial que evolucione (¿Un programa software en sus diferentes versiones?, ¿un ordenador con sucesivas mejoras?).

En biología hay una discusión ya muy antigua que se puede resumir en la siguiente frase "¿la función crea el órgano o el órgano crea la función?". En ella se encierra el secreto del mecanismo de la evolución: cuando los primeros seres dejaron el medio acuático para arrastrarse por la tierra desarrollaron un aparato locomotor adaptado al nuevo medio, ¿fue por la necesidad de andar por lo que se desarrollaron las patas o una predisposición estructural permitió que desarrollaran esa funcionalidad? No vamos a entrar a discutir esta cuestión pero sí nos interesa destacar un aspecto de la misma: la evolución implica la aparición de unas características nuevas cualitativamente diferentes, esto es lo que se denomina emergencia.

La emergencia no se presenta únicamente en la evolución sino que es una cualidad intrínseca de los sistemas. "El todo es más que la suma de las partes" indica que hay algo que no es directamente deducible de la unión de todos los componentes del sistema, esas propiedades que aparecen con el todo son propiedades emergentes.

6.7 Simulación

Para completar esta primera aproximación al estudio funcional de los sistemas es obligado detenernos en la simulación, dado que es una de las herramientas más importantes para estudiar el comportamiento de sistemas. Existen, además, dos conceptos relacionados con la simulación que conviene mencionar brevemente: la construcción de modelos y el análisis de sistemas.

El análisis de sistemas consiste en definir los límites del sistema que se va a modelar, identificar los elementos más importantes y sus interacciones y el entorno de cada uno de ellos, agrupándolos jerárquicamente. Una vez efectuado este análisis se pasa a la modelización, a construir un modelo que refleje los datos obtenidos a través del análisis de sistemas. La simulación es el estudio del comportamiento del

sistema a través de la observación del modelo.

Con la simulación se pueden perseguir objetivos muy diferentes, determinar si las suposiciones sobre el funcionamiento del modelo son válidas (como los modelos utilizados en bioingeniería para el ojo), si los subsistemas identificados se corresponden con una funcionalidad real, efectuar predicciones sobre el comportamiento futuro del sistema, identificar (como la simulación del tiempo atmosférico), estudiar comportamientos erróneos del sistema (como los simuladores utilizados para localizar averías en las líneas telefónicas), etc.

El disponer de un modelo apropiado permite realizar diferentes simulaciones en las que se estudian grupos concretos de variables, se dejan fijos algunos parámetros, se modifica la estructura del sistema, se modifica el peso de las interacciones, etc., lo que permite profundizar el conocimiento que se tiene sobre el sistema.

La simulación es una herramienta muy útil si se utiliza adecuadamente. No se deben confundir nunca los resultados obtenidos en una simulación con los datos que aporta la realidad y siempre se deben tener en cuenta las limitaciones inherentes a la simulación.

7. El sistema, como enfoque global

Hemos visto dos formas de interpretar o estudiar los sistemas, una estructural, atendiendo a su organización y las diferentes partes que lo conforman, y otra funcional, centrada en su comportamiento. Pero estos dos enfoques no se pueden separar radicalmente pues son complementarios. En el estudio de cualquier sistema se puede poner más énfasis en una visión o en otra pero la comprensión total de lo que es el sistema sólo se puede conseguir a través de un enfoque global, construido mediante la adecuada síntesis entre el conocimiento estructural y el conocimiento funcional.

La noción más simple de sistema que podamos pensar ya lleva la idea de generalidad. Considerar un objeto como un sistema es reconocer en él un cuadro de características y una complejidad propia que sólo puede abordarse a través de una interpretación global de ese objeto. El enfoque reduccionista de la visión estructural no nos permite saltar de la interpretación cuantitativa a una visión cualitativa, pero la interpretación funcional, de forma aislada, tampoco nos proporciona los datos necesarios sobre la correlación entre el comportamiento observado y los mecanismos que lo producen. Si queremos realmente conocer un sistema, o actuar sobre él, o construirlo, necesitamos un estudio funcional en el que observemos, comprendamos o definamos el comportamiento del sistema y necesitamos, también, trasladar ese comportamiento a unos mecanismos estructurales que nos permitan actuar, regular o reproducir el sistema.

En el capítulo sobre la historia del enfoque sistémico se puede encontrar esta idea del sistema como enfoque global más desarrollada y en relación con algunas ciencias en las que se ha aplicado con éxito.

8. Resumen

En este capítulo hemos estudiado con cierto detalle la noción de sistema, presentando en primer lugar una relación amplia y detallada de cómo se puede definir. A pesar de que el enfoque sistémico, como veíamos en un capítulo anterior, intenta destacar los aspectos generales de los sistemas, es difícil separar el sistema de su aplicación inmediata y del campo de estudio donde se utiliza. Para superar esta dificultad hemos recurrido a los trabajos de Klir que proporcionan una idea más clara e independiente del dominio de lo que son los sistemas.

La relación de los sistemas y la complejidad aparece plenamente cuando se consideran sistemas abiertos, es decir, sistemas que interactúan con el entorno, adaptándose a él y modificándolo en virtud de esa interacción. Para estudiar los sistemas hemos distinguido dos enfoques, la visión estructural y la visión funcional.

La visión estructural de los sistemas se centra en su organización, los elementos que lo forman, las interacciones entre ellos, las agrupaciones lógicas que se pueden establecer, los límites del sistema y las posibles jerarquías de subsistemas.

La visión funcional hace hincapié en el comportamiento del sistema, en qué hace. Dentro de ella se estudia la entrada y salida al sistema, la realimentación, el equilibrio, no linealidades, sistemas inestables, la evolución, la emergencia de propiedades nuevas y, como herramienta de trabajo para el estudioso de los sistemas, la simulación.

Estos dos enfoques son complementarios y se ha de tener en cuenta a ambos si se pretende llegar a tener una visión global de lo que es el sistema. Este último punto es el que nos sirve de enlace con el capítulo en el que estudiamos la historia del enfoque sistémico, donde aparecen, con una perspectiva histórica, muchas de las ideas que hemos desarrollado algo más formalmente en este capítulo.

9. Bibliografía

Dividida en dos partes. En el apartado de Notas bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Nota bibliográficas

Existen muchos libros en los que se tratan a fondo todas las ideas de este capítulo, muchos de ellos los hemos ya citado en otros capítulos ("El Macroscopio", por ejemplo), sin embargo hay un par de referencias de especial interés:

"Sobre un enfoque de la teoría general de sistemas", artículo de Robert A. Orchard, recogido en el libro, editado por Klir, **Tendencias en la teoría general de sistemas** (Alianza Universidad, número 208). En él se recoge, convenientemente resumido, el trabajo de Klir para construir un método formal del estudio de los sistemas. Con un formalismo matemático apreciable, pero no complicado, este artículo es una buena introducción al estudio formal de los sistemas. Para ampliar aún más se puede consultar **An Approach to General Systems Theory**, de Klir, publicado por Van Nostrand Reinhold, Nueva York, 1969.

Introducción a la dinámica de sistemas, de Javier Aracil (Alianza Universidad Textos, número 58, Madrid, tercera edición, 1987), que, aunque centrado en la dinámica de sistemas según las teorías de Forrester, puede proporcionar una idea bastante clara de una clase de aplicaciones de los sistemas y su estudio funcional a través de simulaciones de ordenador.

Existen unos apuntes donde se recogen muchas de las ideas propuestas aquí. **Sistemas, apuntes para el curso de Fundamentos y Función de la Ingeniería**, son unos apuntes realizados por Sáez Vacas para dicha asignatura y publicados por la E.T.S.I.T. (1976, FFI/SIS 76.01).

Referencias bibliográficas

Aracil, J. (1987), **Introducción a la dinámica de sistemas**, Alianza Universidad, num. 58, Madrid, Tercera Edición.

Ashby, W.R. (1956), **An Introduction to Cybernetics**, John Wiley & Sons, Chichester.

Klir, G.J. (1977) **Tendencias de la Teoría General de Sistemas**, Alianza Universidad, num. 208, Madrid

Orchard (1977), en **Tendencias de la Teoría General de Sistemas**, editado por G.J. Klir, Alianza Universidad, num. 208, Madrid

Rosnay, Joël de, (1977) **El Macroscopio**, Editorial AC, Madrid

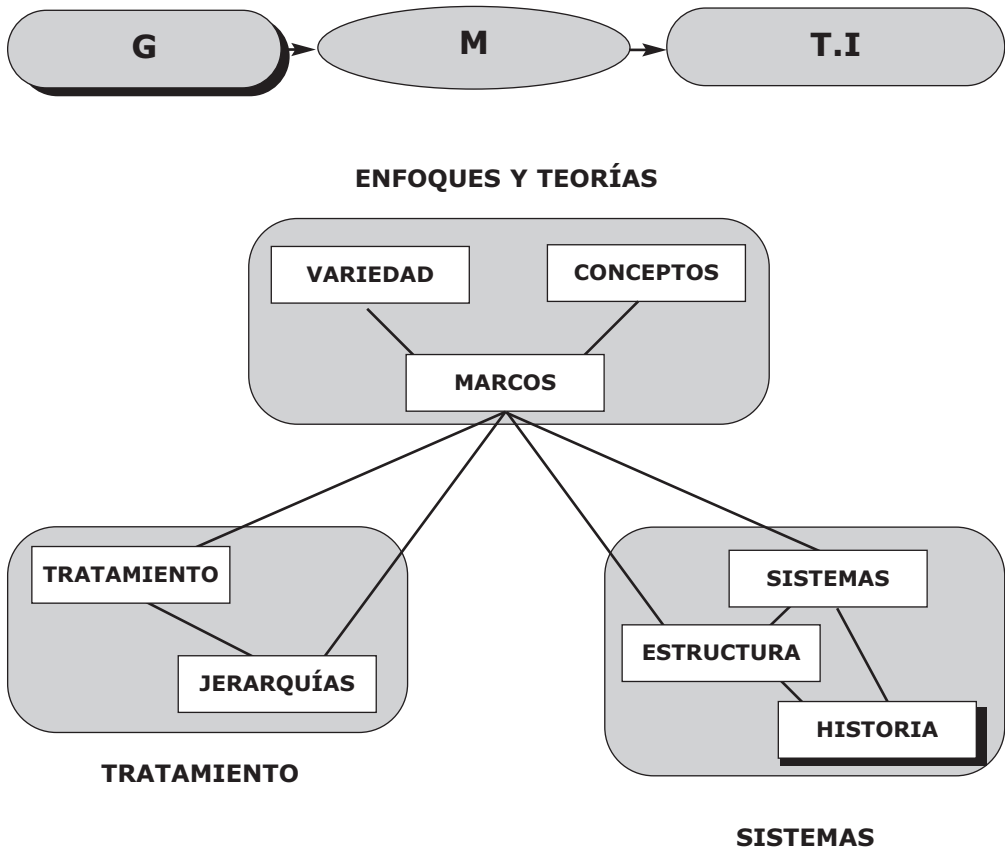
Sáez Vacas, F. y Lampaya, D. "Concepcion multinivélica y cuasidescomponible de sistemas complejos. Aplicación a la informática" **V Congreso de Informática y Automática**, Mayo 1982, pp. 281-286.

Simon, H.A. (1969), **The sciences of the artificial**, MIT Press, Cambridge, Ma.

Capítulo 6: Historia del enfoque sistémico

"Normalmente se dice que los nuestros son unos tiempos de especialización, y es cierto. Pero existe un notable movimiento de convergencia en la ciencia y la enseñanza, que ha tenido lugar especialmente en los cuarenta años que siguieron a la Segunda Guerra Mundial y de forma más acelerada durante los últimos diez años. Están apareciendo nuevos temas, enormemente interdisciplinarios según los patrones tradicionales, y en muchos casos forman la punta de lanza de la investigación. Estos campos interdisciplinarios no siguen el esquema clásico de las ciencias formadas por varias disciplinas; la unión de varios subcampos particulares da lugar a un nuevo campo. El proceso es muy diverso y en constante cambio." Murray Gell-Mann [Pagels, 1989, p. 35]

El movimiento al que se refiere Gell-Mann no es otro que la Teoría General de Sistemas o el enfoque sistémico que si bien ha sido un fracaso en sus presupuestos de partida (construir algo así como una metaciencia), sí ha conseguido logros muy importantes en aplicaciones concretas y ha aportado una nueva perspectiva a la ciencia.



1. Introducción

Pretende ser este capítulo una breve introducción al enfoque sistémico y, por sus características, la forma más apropiada de presentarlo es estudiar sus orígenes y evolución, su historia. En este capítulo nos encontraremos con nombres familiares que ya han aparecido repetidamente en otras partes de estas Notas: Ashby, Wiener, Bertalanffy, Boulding, Beer, etc. y también con otros muchos que no hemos recogido antes pero que son muy representativos para cualquier iniciado en este campo.

El enfoque sistémico es una aproximación multidisciplinar a una serie de problemas que se han querido o creído distinguir en muy diversos campos del saber humano. Este objeto común recibe el nombre genérico de "sistema" y alrededor de él se desarrollan diferentes formas de enfoque sistémico. Como veremos, la aparición de esta corriente de pensamiento, si se le puede llamar así, coincide en el tiempo y en el espacio con otros muchos eventos de primordial importancia para entender el desarrollo de la ciencia y la tecnología que conocemos hoy en día y esto se debe tener muy en cuenta pues es la clave para entender los fundamentos y objetivos de los sistemistas. Y a esto es a lo que nos referimos cuando decimos que la mejor forma de estudiar el enfoque sistémico es comprender su historia.

2. El enfoque sistémico

La base del pensamiento sistémico consiste en reconocer la existencia de una serie de conceptos genéricos aplicables y aplicados en diversos estudios [Rosnay, 1975]. Nociones como la energía, flujos, ciclos, realimentación, sistema abierto, reservas, recursos de comunicación, catalizadores, interacciones mutuas, jerarquías, agentes de transformación, equilibrios y desequilibrios, estabilidad, evolución, etc., son aplicables a la idea genérica de sistema sin entrar en la disciplina concreta ni en el tipo del sistema considerado. La existencia de este vocabulario común en muchos campos parece responder a una aproximación común a los problemas que se encuentran dentro de una misma categoría: la complejidad organizada [Rosnay, 1975] (Weaver, en el capítulo de **Marcos Conceptuales**).

La importancia de este hecho es fundamental pues significa que de ser cierto que existe una aproximación común, se puede establecer una forma de tratar el problema genérico -el sistema- independientemente de la disciplina en la que éste se considere. Esta aproximación común existe y surge de la sinergia entre la biología, la teoría de la información, la cibernética y la teoría de sistemas. A partir de ellas se extraen conclusiones generales, reflejadas en ese vocabulario compartido por todas las disciplinas. Este es el enfoque sistémico y debe verse no como una nueva ciencia, una nueva teoría o una disciplina sino como una nueva metodología que trata de organizar el conocimiento para dar más eficacia a la acción [Rosnay, 1975].

Ésta puede ser la diferencia fundamental entre los que consideran el enfoque sistémico como una herramienta, una metodología que ayuda a interpretar y manejar un mundo complejo y los que consideran que los sistemas son un nuevo paradigma en la ciencia.

Para establecer una relación con algo que nos es más cercano podemos recurrir a la comparación con el enfoque analítico. En él, se pretende desmenuzar la totalidad para estudiar los elementos por separado, aislando interacciones y componentes del resto del todo que forman. Por el contrario, el enfoque sistémico intenta englobar la totalidad de los elementos del sistema estudiado así como las interacciones e interdependencias entre ellos. Por sistema se entiende un conjunto de elementos en interacción y se intenta investigar las invariantes que existen en la interacción de elementos. Esto no es lo mismo que intentar aplicar en otro las conclusiones extraídas para un sistema o de intentar que lo que es válido para un nivel de complejidad lo sea para otro. Esas invariantes son principios generales, estructuras y funcionamiento común a todos los sistemas.

Las diferencias fundamentales entre el enfoque analítico y el enfoque sistémico se recogen en la siguiente tabla [Rosnay, 1975, p. 108]:

Enfoque Analítico	Enfoque Sistémico
Aislado: se centra en los elementos	Relacionado: se centra en las interacciones entre elementos
Considera la naturaleza de las interacciones	Considera los efectos de las interacciones
Se preocupa por la precisión del detalle	Se preocupa de la percepción global
Modifica una variable cada vez	Modifica grupos de variables simultáneamente
Independiente de la duración: los fenómenos considerados son reversibles	Integra la duración y la irreversibilidad
La validación de hechos se realiza por prueba experimental dentro del marco de una teoría	La validación de hechos se realiza por comparación del funcionamiento del modelo con la realidad
Modelos precisos y detallados, pero difícilmente utilizables para la acción (ejemplo: modelos econométricos)	Modelos insuficientemente rigurosos para servir de base al conocimiento, pero utilizables en la decisión y la acción (ejemplo: modelos del Club de Roma)
Enfoque eficaz cuando las interacciones son lineales y débiles	Enfoque eficaz cuando las interacciones son no lineales y fuertes
Conduce a una enseñanza por disciplinas	Conduce a una enseñanza pluridisciplinar
Conduce a una acción programada en detalle	Conduce a una acción por objetivos
Conocimiento de los detalles, metas mal definidas	Conocimiento de las metas, detalles borrosos

3. Orígenes del pensamiento sistémico

Si seguimos los consejos de algún sistemista entusiasta podríamos remontarnos muy atrás buscando precursores a la idea de sistema. Ludwig von Bertalanffy, considerado como el padre de la Teoría General de Sistemas que estudiaremos más adelante, menciona a Aristóteles como el primero que formuló el aserto sistémico fundamental: "el todo es más que la suma de las partes", y liga la noción de sistema al desarrollo de la filosofía europea desde sus más lejanos orígenes [Bertalanffy, 1975, p. 137]. Sin embargo, no es necesario bucear tanto en las profundidades de la historia para trazar un esbozo muy aproximado de las corrientes de pensamiento que dieron lugar al enfoque sistémico. En general, se consideran cinco (Lilienfeld, de quien recogemos la idea, considera también la dinámica de sistemas, pero desde nuestro punto de vista no es relevante en la formación del tipo de enfoque sistémico que nos interesa, por ser muy posterior a éste [Lilienfeld, 1984]):

- **Filosofía biológica**, de Ludwig von Bertalanffy, cuyas ideas cristalizaron más tarde en la Teoría General de Sistemas, representada por la Sociedad Internacional para la Investigación General de Sistemas.
- **Cibernética**, de Norbert Wiener y Ross Ashby, que en sus orígenes se centraba en el estudio de los mecanismos de regulación en los organismos y en las máquinas.
- **Teoría de la información y de las comunicaciones**, de Shannon, Weaver y Cherry que proporcionaron un lenguaje matemático para el manejo de la información y una base formal muy sólida para el estudio de problemas lingüísticos, matemáticos y teóricos relacionados con la transmisión de mensajes
- **Investigación operativa**, de E.C. Williams, originada en Inglaterra durante la II Guerra Mundial e institucionalizada por la Sociedad de Investigación Operativa Americana y la Sociedad de Investigación Operativa de Gran Bretaña.
- **Teoría de juegos**, de Von Neumann y Morgenstern, que además se desarrolla paralelamente a la herramienta básica de los sistemistas: el ordenador.

El enfoque sistémico se confunde a menudo con alguna de estas teorías, principalmente con la Cibernética y con la Teoría General de Sistemas [Rosnay, 1975]. La principal diferencia con la Cibernética es que el enfoque sistémico es mucho más general y la engloba. Mientras la cibernética es la ciencia del control y la regulación, el enfoque sistémico se ocupa de las características invariantes que existen en los sistemas, aunque no cabe duda de que los conceptos cibernéticos son de primordial importancia para entender cierto tipo de sistemas. La diferencia con la Teoría General de Sistemas es quizá más sutil pero también importante. La

T.G.S. (así nos referiremos a ella a partir de ahora) pretende establecer un formalismo matemático para describir el conjunto de sistemas que existen en la naturaleza. El enfoque sistémico propone una forma de ver las cosas pero no una visión tan estricta con la de la T.G.S.

Cibernética

Por su importancia, la Cibernética merece un pequeño comentario aparte. Su origen se puede encontrar en la investigación en física teórica e ingeniería que se llevó a cabo durante la Segunda Guerra Mundial. De ella surgió la idea de la realimentación, que cuando es negativa da lugar a máquinas "autocorrectoras". La hipótesis básica de la Cibernética es que el mecanismo principal del sistema nervioso es la realimentación negativa, hipótesis que se debe principalmente a Ashby y Wiener, aunque fue Wiener el que introdujo el nombre "cibernética" con su libro en 1948. Son varias las ideas nuevas que introdujo la Cibernética tanto en la ingeniería como en la neurofisiología, pero el ejemplo más característico y conocido de lo que se hacía por aquella época es el "homeostato" de Ashby. Consistía en cuatro imanes suspendidos en un campo eléctrico en el que se pueden mover; cada imán recibe corriente de una determinada unidad y desviaciones en un imán producían variaciones en las otras tres fuentes de alimentación. Cada una de estas unidades constaba de un interruptor con 25 posiciones que permitía realizar cambios discretos en la corriente que alimentaba cada imán. De esta forma, un cambio en un imán causaba cambios en la alimentación de los otros tres y estas variaciones, por la misma razón, alteraban su propia alimentación. Ashby afirma que el homeostato simulaba los procesos de aprendizaje en los animales al poder "aprender" por premio o castigo. Utilizando tres selectores y dejando uno libre se fuerza en el homeostato un determinado comportamiento, si falla se altera el interruptor libre (castigo), si acierta, no se toca. Cuando se consigue que el homeostato responda siempre como se le ha indicado se ha conseguido establecer una línea estable de comportamiento. El homeostato permitía además otros muchos experimentos como modificar las realimentaciones, cambiar los cables de un interruptor a otro, forzar respuestas a mano, etc. Es interesante ver estos trabajos del homeostato en relación con la noción de aprendizaje y la, por entonces naciente, inteligencia artificial.

También conviene diferenciar el enfoque sistémico del análisis de sistemas, pues este último es una consecuencia del primero. El análisis de sistemas es una metodología para tratar con sistemas y poder reducirlos a sus componentes e interacciones elementales, pero, para poder hacerlo, primero hay que reconocer los sistemas, que es de lo que trata el enfoque sistémico.

Lo anterior es suficiente para tener una idea más o menos exacta de cómo surgió el enfoque o pensamiento sistémico. Cada autor suele interpretarlo según sus propios intereses haciendo hincapié en algún punto concreto de los que hemos mencionado, principalmente en la Cibernética y en la T.G.S.; pero no son puntos de vista excluyentes sino complementarios pues cada uno recoge las soluciones que se dieron a los mismo problemas desde campos muy diferentes. Aquí desarrollaremos brevemente ambos.

4. Los sistemas, según el M.I.T.

Seguimos en este punto las ideas de Joël de Rosnay [Rosnay, 1975] sobre cómo se desarrolló el enfoque sistémico.

Podemos distinguir tres etapas en la evolución de las ideas sobre sistemas. La primera de ellas se corresponde con la década de los 40 (a finales de la cual aparecen los trabajos de Wiener, "Cybernetics", y la teoría matemática de la información de Shannon) en la que se establecieron diversos principios de control y regulación en las máquinas y se aplicaron en los organismos vivos.

En los años 50 se dio el paso inverso, nociones asociadas a los organismos como memoria, reconocimiento de formas, fenómenos adaptativos, aprendizaje, etc, se empezaron a trasladar al mundo de las máquinas intentando que éstas simularan el comportamiento de los seres vivos. Se produjo también un relanzamiento de los estudios del organismo con progresos importantes en neurología, percepción y visión. (Ejemplo de ello son los trabajos de McCulloch y Pitts sobre la neurona artificial, considerados como la base de las actuales redes neuronales, ver cuadro).



Fig. 1. Las dos primeras etapas de evolución de las ideas sobre sistemas (Rosnay)

En los años 60 estas ideas se extienden a la empresa, la sociedad y la ecología y dan lugar a lo que podríamos llamar enfoque de sistemas generales, que intenta aplicar todos estos conceptos a sistemas de actividades humanas. Hay que mencionar, para ser fieles a la realidad, que estos trabajos son los más cuestionados y criticados de todo el enfoque sistémico por la forma de plantearlos y por las conclusiones a las que llegan. Más tarde volveremos sobre ello. (Como trabajo más representativo de ésta época podemos citar el de Jay Forrester con su Dinámica de Sistemas).

Estas tres etapas constituyen para Rosnay el eje de la evolución del pensamiento sistémico. La razón para elegir como título del apartado "Los sistemas según el M.I.T" es que la mayor parte de todos estos trabajos se hicieron en esa famosa universidad americana.

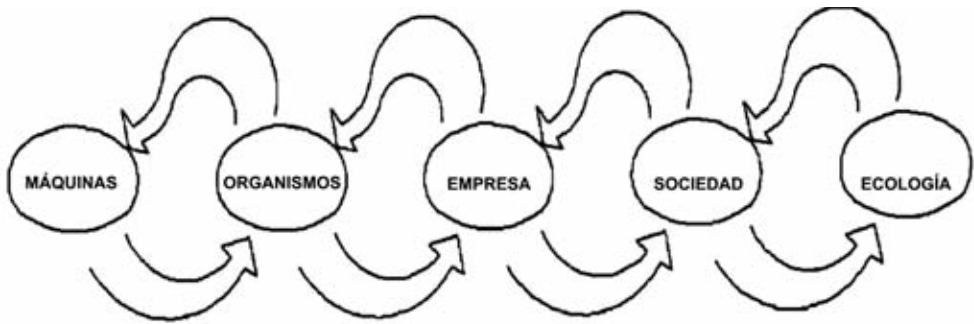


Fig. 2. Extensión evolutiva de las ideas sobre sistemas (Rosnay)

Norbert Wiener, padre de la Cibernética, era un matemático del M.I.T. que, partiendo de trabajos del neurofisiólogo Arturo Rosenblueth y junto con el ingeniero Julian H. Bigelow, a la vista del comportamiento inteligente que parecían exhibir ciertos mecanismos de control (inteligente porque las decisiones se basan en la experiencia y en las previsiones del futuro) llegó a la conclusión de que en el hombre se dan también estos mecanismos de realimentación que permiten controlar la acción y se dedicó a estudiar estos mecanismos de realimentación. En 1948 Wiener publica "Cybernetics " o "Regulación y comunicación en el animal y la máquina". Ese mismo año aparece el trabajo de Shannon y Weaver "Teoría matemática de la comunicación". En esta época se celebran los famosos seminarios de la fundación Josiah Macy, un foro donde se encontraban todas estas ideas y donde se sentaron las bases de muchos avances importantes.

Warren McCulloch, neurofisiólogo que trabajaba con Rosenblueth, tomó estas ideas como punto de partida y empezó a utilizar indistintamente el vocabulario de la ingeniería y de la biología creándose la "jerga" cibernética: aprendizaje, regulación, adaptación, autoorganización, percepción, memoria, etc. Influenciado por Bigelow, McCulloch desarrolla una retina artificial, trabajo en el que colaboran Lettvin, Pitts y Maturana. La necesidad de que las máquinas realicen procesos típicos de los seres vivos acelera la investigación sobre los mecanismos cerebrales. Es el nacimiento de la Biónica, la Inteligencia Artificial y la Robótica.

En esta misma época se funda la Sociedad para la Investigación General de Sistemas, con Bertalanffy, Rapoport, Ashby, Raschewsky y Boulding.

Redes neuronales

La importancia de la sinergia entre los conocimientos de varios campos se puede ilustrar con un ejemplo muy claro. Las ideas desarrolladas para las máquinas se intentan trasladar a la comprensión de los mecanismos cerebrales. El primer trabajo de este tipo fue el de Warren S. McCulloch y Walter Pitts, un neurofisiólogo y un matemático, publicado en 1943 y titulado "A logical calculus of the ideas inmanent in nervous activity". En él plantean un modelo simple de neurona funcionando con una actividad "todo o nada" (similar a la lógica binaria) y una serie de hipótesis que les permiten concluir que toda sentencia de la lógica de proposiciones se puede expresar con una red de neuronas y que existe una sentencia de la lógica de proposiciones para toda red. A partir de este trabajo, Hebb, en 1949, plantea el aprendizaje por correlación y la "regla de Hebb", una expresión aritmética que permite ponderar las conexiones entre neuronas y, por tanto, modificar la red para que aprenda. En 1958 F. Rosenblatt plantea el aprendizaje por imitación y diseña y construye (1962) el Perceptrón Mark I, una retina artificial con una capa de neuronas asociadas que intentaba utilizarse para reconocer formas.

Muchas de estas investigaciones constituyeron la base de las actuales redes neuronales y son una muestra de esa sinergia de conocimientos entre varias disciplinas. Hay que decir también que este primera etapa fue seguida de un época en la que se abandonaron prácticamente las investigaciones, sobre todo a raíz de la publicación en 1969 de un libro de Marvin Minsky y Seymour Papert en el que se subraya la falta de valor científico de los escritos sobre perceptrones, los resultados frustrantes y además se demostraba que con una sola capa de neuronas se obliga a la separación lineal de las características (un problema relacionado con la topología y que indica, en definitiva, que las características, expresadas en un plano han de poder separarse por otro plano). También sugerían la inutilidad de extender estos sistemas a varias capas (aunque luego esa ha sido la solución adoptada ya que se ha demostrado que con tres capas se puede reconocer cualquier espacio de características).

La tercera etapa tiene un representante muy conocido: Jay Forrester. Trabajando en el M.I.T inventa una memoria magnética ultrarrápida de la que posee la mayoría de las patentes y que se utilizó en muchos ordenadores hasta el decenio de los setenta. En 1952 se encarga de coordinar un sistema de alerta y defensa para las fuerzas aéreas con la misión de detectar y rechazar un posible ataque enemigo sobre territorio americano. Durante este trabajo se da cuenta de la importancia del enfoque sistémico. En 1961 crea la Dinámica Industrial, en la que las empresas se consideran como sistemas cibernéticos, en 1964 crea la Dinámica Urbana y en 1971 generaliza todos estos trabajos en la Dinámica de Sistemas y hace un estudio de la dinámica mundial. De esta época es el famoso trabajo para el Club de Roma en el que desarrolló un modelo del mundo del que se extrajeron conclusiones que todavía hoy se discuten.

Dinámica de Sistemas

La dinámica de sistemas es una metodología para el estudio, modelado y simulación de sistemas dinámicos, es decir, de sistemas en los que lo que interesa es su evolución con el tiempo. Como metodología se basa en las técnicas tradicionales de gestión de sistemas sociales, la teoría de sistemas realimentados y la simulación por computador [Aracil, 1979]. Los conceptos que se utilizan para modelar los sistemas son los niveles (estados), flujos (variaciones de los estados), canales materiales y de información (que indican transmisiones de una magnitud física o de información), variables auxiliares y exógenas (para simular ciertos aspectos de los sistemas), constantes (partes invariables del modelo) y retardos (elementos que simulan un retraso en la transmisión de información o de material). En la simulación del comportamiento de estos sistemas se utiliza un pequeño aparato matemático que sirve para reflejar las variaciones de flujo, la influencia de las variables y las constantes, etc. En general la modelización es bastante intuitiva aunque a base de repetir estas estructuras se llegan a conseguir modelos realmente complejos. Para la programación del ordenador para simulación se utiliza un lenguaje específico de programación llamado DYNAMO.

Con estas herramientas Jay Forrester elaboró los conocidos modelos del mundo encargados por el Club de Roma y que le sirvieron a este último para realizar una serie de predicciones sobre la población mundial, la evolución de los recursos naturales, la contaminación, calidad de vida, etc. Estos modelos han sido ampliamente comentados y criticados no sólo en los aspectos relacionados con lo puramente científico sino también desde el punto de vista político y de los objetivos perseguidos. Con ellos se pone de manifiesto un peligro evidente de la metodología sistémica que se hace aún más patente en la dinámica de sistemas. El modelador del sistema refleja en el modelo que construye sus propias creencias de forma que el sistema no refleja exactamente la realidad sino cómo la ve el que la modela, sobre todo cuando se trata de sistemas tan poco conocidos y predecibles en su comportamiento como es el mundo.

5. La Teoría General de Sistemas

Se considera fundador de la teoría a Bertalanffy, por su insistencia en la creación de un cuerpo teórico partiendo de todas las ideas que iban apareciendo en su momento sobre sistemas en diferentes campos y que consideraba podían agruparse bajo una única disciplina. Sus formulaciones relacionadas con el concepto de sistema abierto fueron las primeras en introducir la idea de sistema como un movimiento científico, basándose primordialmente en la biología donde sostenía que el problema fundamental era encontrar las leyes de sistemas biológicos donde hay subordinación de las partes y los procesos componentes. Conceptos como el orden, la regularidad y el automantenimiento paralelos al cambio continuo y la regulación son muy difíciles de explicar desde la física o las matemáticas y para Bertalanffy sólo podrían explicarse utilizando nuevos marcos conceptuales [Lilienfeld, 1984].

En 1947 Bertalanffy afirmaba: "existen modelos, principios y leyes aplicables a sistemas generalizados o a subclases cuyas independientemente de su naturaleza, del carácter de los elementos componentes y de las relaciones o "fuerzas" existentes entre ellos. Postulamos una nueva disciplina llamada Teoría General de Sistemas" [cita recogida en Bertalanffy, 1979, p. 15]. Esta Teoría General de Sistemas surge, según Bertalanffy, de las siguientes consideraciones [citadas en Aracil, 1979]:

- a) Existe una tendencia general hacia la integración en todas las ciencias, tanto naturales como sociales;
- b) Esta integración puede centrarse en una teoría general de sistemas;
- c) Esta teoría puede ser un medio importante para conseguir una teoría exacta en los campos no físicos de la ciencia;
- d) Esta teoría conduce a la unidad de la ciencia, al desarrollar principios unificadores que integran, verticalmente, el universo de las ciencias individuales;
- e) Todo ello puede conducir a una integración, ampliamente necesitada, en la educación científica.

Y sobre estos puntos intentó construir una metateoría de alto nivel sobre sistemas. En 1950, Bertalanffy publica "An Outline of General Systems Theory" (The British Journal for the Philosophy of Science, 1 (1950), pp. 134-165) y cinco años más tarde aparece en la misma publicación (1955, p. 331) un anuncio de que se está creando una Sociedad para el Progreso en Teoría General de Sistemas. Algunos de los promotores de la idea eran Bertalanffy, biólogo, K.E. Boulding, economista, R.W. Gerard, fisiólogo, y A. Rapoport, matemático. El propósito era la promoción

de "sistemas teóricos aplicables a más de un campo de los tradicionalmente considerados en la ciencia". Y los objetivos eran los siguientes [Checkland, 1981]:

- a) Investigar el isomorfismo de conceptos, leyes y modelos en varios campos y facilitar la transferencia de conocimientos de un campo a otro;
- b) Promover el desarrollo de modelos teóricos adecuados en las áreas que carezcan de ellos;
- c) Suprimir la duplicación de esfuerzos teóricos en diferentes campos;
- d) Promover la unidad de la ciencia a través de la mejora de las comunicaciones entre especialistas.

Para los seguidores de Bertalanffy la Teoría General de Sistemas es, en última instancia, una perspectiva o paradigma nuevo, una nueva forma de hacer ciencia. E incluso van más allá al considerar que más que una teoría, en el sentido tradicional del término, es un paradigma para desarrollar teorías y síntesis transdisciplinarias.

Esta es la diferencia fundamental entre la T.G.S. y el enfoque sistémico. La primera busca una formalización bastante estricta de los conceptos e ideas asociados a los sistemas y cree en la existencia de una teoría de sistemas por sí misma sin relación con ningún campo concreto. Actualmente existen discrepancias dentro de los sistemistas de la T.G.S. sobre cómo interpretar el nombre de la sociedad que les acoge, International Society for General Systems Research puede referirse a la investigación de sistemas generales, es decir al estudio de cualquier tipo de sistema, o la investigación general de sistemas, que constriñe en cierta forma el campo de actuación. Los partidarios de la primera opción llevan la idea de sistema al campo social, a la psiquiatría, a la psicología y a la filosofía (a la que Bertalanffy dedica muchos de sus escritos) y, como se les ha criticado con frecuencia, hacen de ello una ideología cerrada. Los partidarios de la segunda opción tienden a un estudio mucho más formal y matemático de los sistemas buscando aplicaciones concretas no tan difusas y de resultados no tan inciertos. Trabajos como los de Klir (ver **Marcos Conceptuales**), Checkland, Beer, etc., persiguen una aplicación inmediata de la noción de sistema.

En cuanto al enfoque sistémico, se trata más de una forma de interpretación, de una herramienta, que de una disciplina científica. Sus seguidores no lo proponen como un nuevo paradigma científico aunque reconocen la importancia de este nuevo punto de vista. Para la mayoría de ellos el enfoque sistémico es una herramienta conceptual que permite manejar realidades complejas y que además es un reflejo de esa realidad. Otra cosa es que dispongan de métodos genéricamente válidos para tratar con los sistemas o que se haya alcanzado un grado de formalización aceptable. Para terminar este apartado podemos citar a Simon, que resume

muy bien el papel que juega el enfoque sistémico: "su popularidad es más la respuesta a una acuciante necesidad de sintetizar y analizar la complejidad que el desarrollo de un cuerpo de conocimientos y técnicas para tratar la complejidad" [Simon, 1969].

6. Logros del enfoque sistémico

No cabe ninguna duda de que los propósitos de los que parte la T.G.S. son muy interesantes pero no es menos cierto que, hasta ahora, tan loables objetivos no se han conseguido. Hay que decir que, en puridad, tampoco existe la teoría general que buscaba Bertalanffy y los mismos fundamentos de la Teoría General de Sistemas han sido objeto de diversas críticas. Para muchos, no es siquiera un cuerpo coherente de conocimientos, sino una serie de analogías muy difíciles de formalizar matemáticamente. La falta de contenido es el precio que se paga por la búsqueda de una generalidad absoluta.

Es cierto que a pesar del tiempo transcurrido no se ha llegado a un cuerpo coherente de conocimientos y lo que se sabe está lejos de poder considerarse una nueva disciplina científica o un paradigma revolucionario. Sin embargo, el enfoque sistémico ha facilitado importantes aportaciones en muchos campos concretos. Ya vimos antes cómo durante el desarrollo de gran parte de estas ideas y como resultado de la interacción entre varias disciplinas surgieron trabajos interesantísimos que hoy se consideran la base de muchas disciplinas importantes.

Los trabajos en el M.I.T. además de crear una terminología hoy ampliamente utilizada, proporcionaron las bases para la Inteligencia Artificial y la Robótica. Ya hemos mencionado los trabajos de McCulloch, pero no fueron los únicos porque todo este movimiento contribuyó a crear un estado de opinión frente a una serie de problemas y esto se reflejó en muy diversas disciplinas desde la neurofisiología a la ingeniería.

La cibernética, a partir de Wiener y Ashby, proporcionó las herramientas básicas de los actuales servomecanismos y estudió por primera vez uno de los conceptos más importantes en el control de máquinas: la realimentación. Como ejemplo que merece comentario aparte podemos citar los trabajos de Beer sobre sistemas viables, basados íntegramente en la cibernética, que hoy en día se aplican con éxito en la gestión de empresas e incluso se ha llegado a diseñar sistemas de gestión de naciones.

La dinámica de sistemas de Forrester, a pesar de las muchas críticas a sus modelos del mundo, también se ha convertido en una herramienta importante para simulaciones no sólo en ingeniería sino en los más diversos campos. Existen varios lenguajes de programación, el primero y más conocido de todos es DYNAMO, especialmente concebidos para simular este tipo de sistemas en ordenador.

En general han sido muchas las disciplinas que se han beneficiado del enfoque sistémico y existen muchos trabajos que aunque sin reconocerlo explícitamente recogen la idea de sistema o emplean alguna de las herramientas de los sistemistas. Se puede decir que ha propiciado importantes avances en el desarrollo de conceptos y teorías menos generales de sistemas aplicables a problemas específicos o a determinadas clases suficientemente amplias de problemas.

7. Crítica a la noción de sistema

A pesar de considerarse, en general, como una aportación importante, el enfoque sistémico, y más la T.G.S., ha sido objeto de diversas críticas, entre ellas la de ser una mezcla de ideas tomadas de muchos sitios y sin un esfuerzo por cohesionarlas:

"...el enfoque, tal como lo encontramos ahora, recuerda tanto por su historia como por su constitución al fenómeno conocido como "puding de Roxbury". Esta formación se encontró en un suburbio de Massachusetts, fue producto de un movimiento glacial, el cual durante centurias arrastró, acumuló y luego incorporó un tipo de rocas bastante heterogéneo, todas colocadas en una matriz y luego solidificadas en una masa aglomerada. Muchos fragmentos conservan todavía su identidad y su carácter original, algunas han sufrido metamorfosis en diverso grado. De igual manera, el enfoque de sistemas es una especie de mosaico, hecho de trozos y piezas de ideas, teorías y metodologías de un cierto número de disciplinas, entre las cuales son discernibles -además de la ingeniería- la sociología, biología, filosofía, psicología y economía.

Cada disciplina tiene su propia concepción intrínseca y fundamental de sistema, conjuntamente con sus propias definiciones, principios, supuestos e hipótesis. Pero hay una dinámica que las empuja conjuntamente, que las hace "gemütlich" y las provee de un parentesco mutuamente sustentador. Este consiste en su orientación y énfasis en la totalidad de la experiencia, entidad o fenómeno bajo consideración...". (Ira R. Ross, "Systems Analysis in Public Policy: A Critique", p. 27, Berkeley Calif. Univ. of Calif. Press, 1977, [citado por Lilienfeld, 1984, p. 275]

La crítica al movimiento sistemista ha sido muy fuerte desde muchos campos y generalmente se basa en la falta de una formalización seria de los conceptos que maneja más allá de las aplicaciones concretas. Lilienfeld apunta 4 "vicios dominantes" en los sistemistas [Lilienfeld, 1984]:

1. Una debilidad por las propuestas pragmáticas asociada a una escasez de resultados concretos.
2. Una inclinación por fórmulas abstractas, esquemáticas y diagramas que tienen poca aplicación práctica.

3. Peticiones de principio fundamentales que toman las formas de un cambio irregular y presumiblemente invisible desde los "sistemas" concretos en toda su plenitud y complejidad hasta los modelos formales cerrados que se basan en supuestos simplificadores.
4. Ausencia de trabajos concretos más allá del refinamiento del sistema mismo.

Y además añade los siguientes tres puntos como características de la teoría sistémica [Ibid., p.290]:

1. Es una analogía, pese a la negación de muchos teóricos de sistemas.
2. Con la teoría de sistemas se desarrollan pocas operaciones, a excepción de la teoría de la comunicación, donde la filosofía se derivó después del hecho; la filosofía misma no permite operaciones.
3. Asume un determinismo en la ciencia que muchos científicos rechazan.

Críticas que son ciertas en muchos casos y que en parte se derivan de la insistencia de algunos sistemistas en considerar a los sistemas como una disciplina científica y un nuevo paradigma. Como cuerpo genérico de ideas es totalmente cierto que la sistémica ha dado muy pocos frutos, pero si vamos a la aplicación de esas ideas en campos concretos la situación es totalmente distinta. En el campo de las ciencias "duras" los sistemas se han aplicado con bastante éxito y hay también aproximaciones interesantes a problemas "blandos" que ciertamente son más propuestas de actuación o interpretación que aplicación práctica. Donde la crítica aparece más justificada es en campos como la sociología o la filosofía, en las que es frecuentemente cierto que existe una auténtica ideología y propuestas más que discutibles.

Rosnay también señala, con ánimo menos crítico pues es uno de los valedores del enfoque sistémico, algunos peligros de los sistemas [Rosnay, 1975, pp. 126-127]. En primer lugar apunta que es necesaria una desmitificación, que está en la línea de apartarse de los sistemistas "puros y duros" como Bertalanffy y adoptar más una actitud de interés transdisciplinar, en la complejidad y en la interdependencia. Una visión global que permita percibir la realidad compleja de las cosas y no un reduccionismo que tome como base el propio método sistémico. También aconseja huir de una aplicación sistemática de los principios sistémicos, lo que resulta obvio pues el enfoque analítico y el enfoque sistémico son complementarios, y además puede darse el caso de que una utilización abusiva de analogías, homologías e isomorfismos compliquen las cosas en lugar de aclararlas. La unificación total que persiguen los teóricos de la T.G.S. corre el riesgo de convertirse en una simplificación abusiva, una idea fija o una receta de pensar (aquí Rosnay cita a Edgar Morin).

Es cierto que los matemáticos y físicos son los que más problemas tienen para adaptarse a la idea de sistema pues la base fundamental de su pensamiento es el método analítico. Los biólogos, médicos y economistas tienen más facilidad para percibir la noción de sistema pues continuamente están tratando con interacciones y con temas a los que es difícil aplicar el método analítico. Pero es peligroso intentar llevar las conclusiones de un campo a otro sin una profunda reflexión (recordar los postulados de la T.G.S.). De la misma forma, perseguir una teoría unificada de los sistemas que recoja todos sus aspectos e implicaciones y sea genéricamente aplicable tiene unos riesgos grandes de apartarse de la realidad al ir tras un formalismo matemático generalizador que no se corresponda con la realidad.

8. Resumen

El enfoque sistémico ha de verse como una herramienta intelectual muy poderosa que responde, como dice Simon, a una necesidad concreta de tratar la complejidad. Para comprender su significado y valorarlo adecuadamente ha sido interesante repasar su historia, jalonada de descubrimientos y avances importantes, y aceptar las críticas que se le hacen cuando se intentan llevar estas ideas a extremos poco razonables.

Es difícil ver el enfoque sistémico o la Teoría General de Sistemas como una disciplina científica, a pesar de los muchos sistemistas que lo sostienen. Los conceptos son, necesariamente, demasiado vagos e imprecisos pues lo que se busca es la generalidad. Aquí es donde se corre el riesgo de no decir nada sobre todo (frente al riesgo de la especialización, saber todo sobre nada) y donde muchas veces se han centrado las críticas al intentar presentar las ideas sobre sistemas no como una interpretación que permite manejar la complejidad sino como una teoría completa y formal de los sistemas. La propia idea de sistema es bastante difusa pues en cada aplicación se define el sistema de nuevo y aunque hay una concepción básica común las diferencias son a veces importantes.

No cabe duda, sin embargo, de que el enfoque sistémico es una gran ayuda para comprender sistemas en los que las interacciones son muy fuertes e importantes, en contraposición a los sistemas que estudia la física clásica en los que la separabilidad y el principio de superposición son aplicables, por eso es por lo que muchos de los sistemistas son biólogos y también es que muchas de las críticas vienen de matemáticos y físicos que se quejan de la falta de formalismo y precisión de las ideas sistémicas.

Desde el punto de vista de la ingeniería, el enfoque sistémico, como "software mental", es muy útil para percibir la riqueza y la complejidad de los diseños y los desarrollos y, al mismo tiempo, nos proporciona una serie de herramientas básicas para tratar esa complejidad y crear nuestra propia metodología de sistemas.

9. Bibliografía

Dividida en dos apartados. En el apartado Notas Bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

La bibliografía existente sobre sistemas es muy abundante, algunas de las referencias que hemos utilizado son las siguientes:

El Macroscopio, de Joël de Rosnay, [Editorial AC, 1977, Madrid]. Un libro con propuestas realmente audaces, muy interesante y ameno de leer. Toca muchos temas y los ejemplos que utiliza van desde la biología (la célula) a la economía, por lo que es muy útil para ver la aplicación del enfoque sistémico a diversas disciplinas. Capítulos especialmente interesantes son los dos primeros, con ejemplos de aplicación del enfoque sistémico a la ecología, la economía, la ciudad, la empresa, el organismo y la célula, y una buena introducción a la historia del enfoque sistémico (aunque algo sesgada hacia los sistemas dinámicos, dejando de lado la T.G.S.). La publicación original, en francés, data de 1975.

Tendencias de la Teoría General de Sistemas, de George J. Klir [Alianza Universidad, 208, Madrid 1978]. Referencia frecuente (en el capítulo dedicado a la **Simplificación** volverá de nuevo), en este libro se recogen varios artículos de diversos autores y especialistas en sistemas, desde Bertalanffy a Ashby, Weinberg y el propio Klir y es muy interesante para profundizar en la problemática asociada a los sistemas, su aplicación en diferentes campos y las líneas de estudio seguidas por destacados sistemistas.

Perspectivas en la Teoría General de Sistemas, de Ludwig von Bertalanffy, en la misma colección que el anterior (número 230) y que puede ser útil para ver cuáles eran los planteamientos y motivaciones iniciales que llevaron a la creación de la Teoría General de sistemas. En este libro hay, quizás, un énfasis filosófico y doctrinal que puede dificultar la lectura, por lo demás muchas de las ideas que se exponen son las más criticadas del movimiento sistémico.

Teoría de Sistemas, de R. Lilienfeld, [Editorial Trillas, México 1984], traducción del inglés un tanto "especial", lo que puede dificultar en algún momento la lectura, pero que no desmerece en absoluto el interés del libro. Una crítica muy fuerte, y razonada, de ciertos aspectos del enfoque sistémico, sobre todo de los doctrinales y su aplicación como ciencia social. La primera parte contiene una reseña muy elaborada de todas las corrientes de pensamiento y autores concretos que destacan en el movimiento sistémico.

Perspectivas en la revolución de los ordenadores, de Z.W. Pylyshyn, [Alianza Universidad, No. 119, Madrid 1975]. Un auténtico "clásico", de él hemos extraído algunas referencias sobre la cibernética y puntos concretos que hemos ido tocando a lo largo del capítulo. No es un libro dedicado a los sistemas sino al desarrollo histórico de la ciencia de los ordenadores. Fundamental para cualquiera que quiera profundizar en la informática y los ordenadores. Se compone de bastantes artículos, extensos en su mayoría, algunos de los cuales fueron aportaciones básicas, entre ellos destacan los de Von Neumann, Babbage, Shannon, Turing, Simon y Newell.

Referencias bibliográficas

Aracil, J. (1987) **Dinámica de Sistemas**, Alianza Universidad, num. 58. Madrid, tercera edición.

Bertalanffy, Ludwig von (1979), **Perspectivas en la Teoría General de Sistemas**, Alianza Universidad, número 203, Madrid.

Checkland, P, (1981) **Systems Thinking, Systems Practice**, J. Wiley & Sons, Chichester, 1981.

Klir, G.J. (1978) **Tendencias de la Teoría General de Sistemas**, Alianza Universidad, No. 208, Madrid.

Lilienfeld, R. (1984) **Teoría de Sistemas**, Editorial Trillas, México.

Pagels, H.R. (1989) **The Dreams of Reason, the computer and the rise of the sciences of complexity**, Bantam Books, N.Y.

Pylyshyn, Z.W. (1975), **Perspectivas en la revolución de los ordenadores**, Alianza Universidad, No. 119, Madrid.

Rosnay, J. de (1977) **El Macroscopio**, Editorial AC, Madrid (Traducción de F. Sáez Vacas).

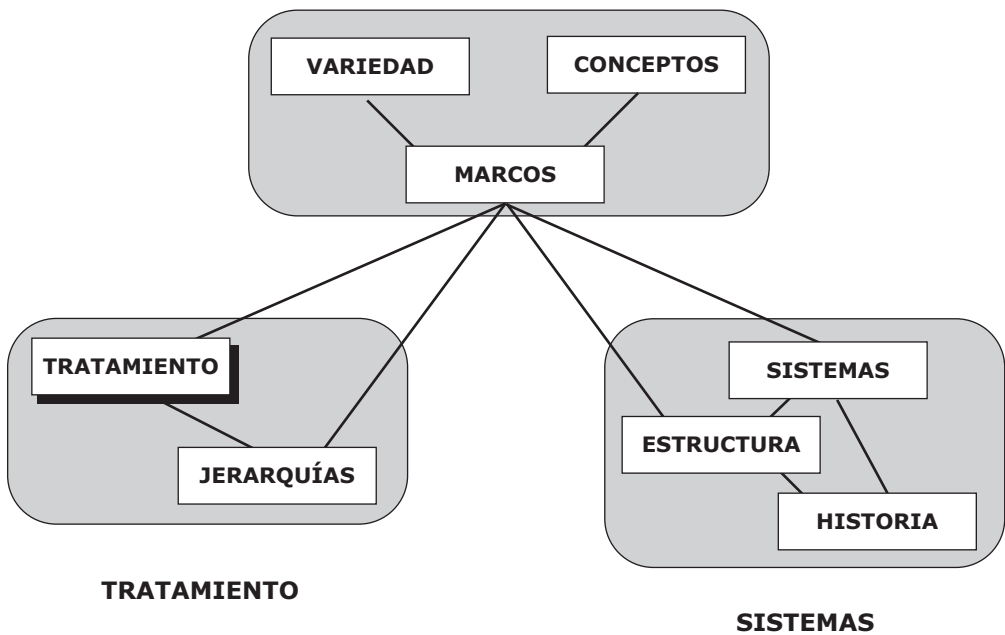
Simon, H.A. (1969) **The Sciences of the Artificial**, MIT Press, Cambridge Ma.

Capítulo 7: Tratamiento de la complejidad: simplificación

La simplificación es un concepto indisoluble de la complejidad. Si se admite la existencia de una cierta complejidad en los sistemas o en los objetos, hay que pensar en formas de reducirla para poder abordar los problema asociados. Cualquier proceso de diseño es un proceso de tratamiento de complejidad en el que las ideas se van refinando sucesivamente hasta llegar a un modelo más o menos simplificado de lo que es la realidad. Dado que la simplificación es inevitable en todo proceso epistemológico, es conveniente analizar cómo se puede realizar preservando la mayor cantidad posible de significado y relevancia del problema.



ENFOQUES Y TEORÍAS



1. Introducción

Una vez reconocida la existencia de la complejidad en los sistemas y en el mundo en general hay que tratar de establecer las formas de tratarla. La simplificación, más que una herramienta, es una necesidad, un enfoque o un planteamiento imprescindible para tratar la complejidad. Sin embargo el problema de la simplificación no se reduce a la eliminación de ciertas características o a despreciar algunas, o muchas, interrelaciones. El objetivo de este capítulo es presentar el problema general de la simplificación y las diversas aproximaciones a él que han adoptado diversos autores con la idea de mostrar todas las facetas implicadas a la hora de simplificar un sistema. Muchas de las ideas que veremos aquí están íntimamente relacionadas con las presentadas en otros capítulos y se deben a la estrecha relación que existe entre la forma de considerar los sistemas y la aproximación que se elige para simplificarlos.

2. El problema general de la simplificación. George J. Klir

La complejidad de los sistemas se estudia principalmente para desarrollar métodos con los cuales se pueda reducir a niveles manejables la complejidad de sistemas incomprensibles o intratables. Esta es la definición más genérica que se puede hacer de la simplificación y, al mismo tiempo, nos indica la relación que existe entre la simplificación y el estudio de la complejidad de los sistemas. Para Klir, los problemas de simplificación son quizás los problemas más importantes del estudio de los sistemas.

Quizá convenga recordar aquí que Klir parte de la idea de que la noción de complejidad está directamente relacionada con la de sistema. Y él desarrolla el tratamiento de la simplificación de acuerdo con los principios generales de complejidad que propone y su jerarquía de sistemas (ver el capítulo de **Marcos Conceptuales**).

Principios generales de la complejidad de los sistemas

- 1. La complejidad de un sistema, sea del tipo que sea, debe ser proporcional a la cantidad de información necesaria para describir el sistema.***
- 2. La complejidad de un sistema debe ser proporcional a la cantidad de información necesaria para resolver cualquier incertidumbre asociada con dicho sistema.***

Para Klir, la complejidad de cualquier tipo de sistema está sujeta a dos principios generales, uno que la relaciona con la cantidad de información descriptiva necesaria y otro que la relaciona con la incertidumbre existente en el sistema. La simplificación debe estar, por tanto, orientada a reducir la complejidad basada en la información descriptiva y la basada en la incertidumbre. El principal problema radica en que estas dos complejidades son, en cierta forma, contrapuestas, es decir, cuando reducimos una, aumenta la otra o, en el mejor de los casos, permanece inalterable.

Esta relación es fácil de ver con el siguiente ejemplo: definamos una clase de una universidad como un conjunto de alumnos que atienden a las mismas asignaturas en el mismo horario y en la misma aula. Este conjunto tiene una complejidad descriptiva asociada y una complejidad relacionada con la información necesaria para resolver las incertidumbres que se nos presentan respecto a él. Una definición de una clase determinada que reduce mucho la complejidad descriptiva es la siguiente: esta clase está formada por 90 alumnos. Es una definición buena y la complejidad descriptiva es pequeña, pero la complejidad dependiente de la incertidumbre está prácticamente en su grado máximo, con esa definición no podemos resolver prácticamente ninguna incertidumbre.

Supongamos que queremos estudiar la preferencia que tiene esa clase por un determinado equipo de fútbol. Saber que la clase tiene 90 alumnos nos proporciona la incertidumbre máxima (no sabemos nada), en este caso la complejidad descriptiva es muy pequeña y la complejidad de incertidumbre es muy grande. Si aumentamos la complejidad descriptiva y decimos que la clase está formada por 90 alumnos, de los cuales 15 son de Barcelona, 20 de Vigo y 55 de Madrid, la incertidumbre sobre las preferencias futbolísticas de esa clase disminuye en una medida importante pues ya nos permite suponer, bastante razonablemente, cuales van a ser las preferencias de los tres grupos en que se ha dividido la clase. Es importante ver y entender cómo para reducir la incertidumbre hay que aumentar la complejidad descriptiva y cómo una disminución de esta última complejidad conduce a un aumento de la incertidumbre.

Para Klir, todas las estrategias de simplificación pueden reducirse a una formulación general, que no trataremos aquí. Las diferencias que hay entre unos métodos de simplificación y otros se pueden resumir en cuatro puntos:

1. El tipo epistemológico de sistema que se simplifica: Una forma genérica de simplificación es eliminar variables. Según sea el tipo de sistema con el que se trabaje así será la eliminación que se haga. Por ejemplo, en un sistema fuente se pueden eliminar directamente variables o reducir el nivel de resolución de sus valores, con lo cual se simplifica la complejidad descriptiva, o se puede reducir el número de estados posibles de un sistema generativo para reducir así la complejidad dependiente de la incertidumbre. Como ejemplo de un

sistema fuente podemos tomar un circuito electrónico en el que, para reducir la complejidad descriptiva, no consideramos la disipación de potencia y nos quedamos sólo con las variables de intensidad de corriente y tensión. Para reducir la complejidad relacionada con la incertidumbre podemos pensar en reducir el ruido para evitar comportamientos extraños o en limitar la entrada para evitar que algún componente entre en saturación.

2. El conjunto de simplificaciones consideradas válidas dentro de un sistema: es evidente que no en todos los sistemas se puede aplicar el mismo tipo de simplificaciones. Así, en los sistemas de los que se ocupa la mecánica clásica se puede utilizar el principio de superposición para simplificar la complejidad descriptiva, esto es absolutamente impensable en un sistema social, lo que determina una estrategia de simplificación radicalmente distinta, un psicólogo se sentiría muy feliz si pudiera estudiar las relaciones de grupos dividiéndolas en interacciones dos a dos. El éxito de los mecanismos de simplificación depende mucho de ligerísimas variaciones en el sistema considerado y lo que puede valer para un caso deja de servir para otro.
3. Los dos tipos de complejidad determinados por los principios generales: Dependiendo de la estrategia que elijamos, simplificaremos la complejidad descriptiva o la relacionada con la incertidumbre y la conveniencia de reducir una u otra depende de cada caso particular. El método más común de reducir la complejidad descriptiva de un sistema es descomponerlo en subsistemas, lo que conlleva un aumento (o, en el caso mejor, una estabilización) de la complejidad dependiente de la incertidumbre, dado que el sistema es más que la suma de las partes. Un ejemplo útil para ver mejor este punto es la célula. Al describirla, la dividimos en diversas unidades funcionales, cada una con una misión concreta. Sin embargo, la reunión simple de todos estos elementos no da lugar a una célula, simplemente a un conjunto de elementos. Al dividir la célula en partes funcionales diferenciadas hemos dejado de lado todas las interrelaciones entre estas partes, interrelaciones que son fundamentales para el funcionamiento del todo pero que no son relevantes en el comportamiento de cada parte. Por otro lado, los sistemas presentan la propiedad de la emergencia, es decir, que el todo presenta características que no son explicables a partir de los elementos que lo forman. Estas características, al quedar eliminadas descomponiendo la célula en subsistemas, aumentarán la incertidumbre a la hora de predecir el comportamiento de la célula. Un método muy en boga para reducir la complejidad relacionada con la incertidumbre es disminuir la precisión exigida y utilizar la teoría de conjuntos borrosos que permite manejar un grado de incertidumbre mayor. Evidentemente, introducir la borrosidad en un sistema complica mucho su descripción y por tanto aumenta la complejidad de este tipo.

4. La naturaleza de otros órdenes de preferencia (respecto a las simplificaciones).

Con todo ello, Klir nos presenta una formulación muy amplia de problema de la simplificación que nos es muy útil como punto de partida para desarrollar este tema.

3. La simplificación, según Gerald M. Weinberg

Para Weinberg la simplificación es un punto de mucha importancia, tanto es así, que define la ciencia de los sistemas como la ciencia de la simplificación. Para ilustrarlo propone el siguiente ejemplo:

Consideremos el problema de describir un sistema de dos objetos. En este sistema hay cuatro ecuaciones: a) cómo se comporta cada objeto por separado, dos ecuaciones; b) cómo se relacionan entre ellos, una ecuación; y c) qué sucede cuando los objetos no están, una ecuación, conocida como ecuación de campo. Para describir un sistema con n cuerpos se tiene 2^n ecuaciones. Para 10 cuerpos esto resulta en más de 1.000 ecuaciones. Si lo que se pretende es estudiar el sistema solar, formado por unos 100.000 cuerpos el número de relaciones es de $10^{30.000}$, un problema realmente intratable.

Hipotéticamente, éste fue, sin embargo, el problema que se le planteó a Newton cuando intentó describir el comportamiento del sistema solar. Resulta evidente que es necesario utilizar algún medio de simplificación que haga que el problema se reduzca a límites manejables:

En primer lugar consideremos que podemos prescindir de las masas pequeñas ya que no influyen de manera decisiva en el comportamiento de las masas grandes como son los planetas o el sol. Con ello hemos reducido el problema de $10^{30.000}$ ecuaciones a unas 1.000, correspondientes a 10 cuerpos. La diferencia es apreciable pero, no disponiendo de un ordenador, 1.000 ecuaciones se pueden plantear pero no resolver.

El segundo paso es aplicar el principio de superposición, es decir, suponer que se pueden aislar las relaciones de los cuerpos dos a dos, lo cual es factible porque la ley de la gravedad así nos lo indica ya que postula que la fuerza de atracción entre dos cuerpos no depende de la presencia de un tercero. Con ello podemos reducir nuestro sistema de ecuaciones a unas 45. Sistema que ya se puede considerar resoluble.

Pero Newton fue más allá. Supuso que el sol, al ser mucho mayor que el resto de las masas planetarias, era la masa dominante en el sistema, de forma que se podrían ignorar las relaciones entre los planetas, quedándose sólo con las relaciones sol-planeta. Esto reduce el sistema de ecuaciones a 10. Y aún hizo más, supuso

que las relaciones sol-planeta podían aislarse una a una, de forma que la resolución de las ecuaciones requería mucho menos cálculo pues n ecuaciones requieren n^2 unidades de computación, mientras que n ecuaciones separadas requieren únicamente n unidades de computación. En este punto dejó de simplificar y ya resolvió el sistema.

El viento del sol

En cierto sentido se puede decir que los vuelos espaciales se deben a Newton. Gracias a sus cálculos, a su ley de la gravedad y a sus teorías se pueden calcular las órbitas de los satélites. Sin embargo, el método seguido por Newton tiene simplificaciones peligrosas para los que calculan una determinada órbita para un determinado satélite pues a menudo estas simplificaciones no son del todo correctas. Ejemplo de ello son los cálculos de la órbita del satélite Echo (Eco, en español), que era, básicamente, una esfera de Mylar inflada. Tras meses de trabajo se vio que las ecuaciones clásicas no valían para el Echo. Se daba el caso, averiguado tras muchos cálculos, de que el Echo, dada su composición, tenía una densidad muy baja y comparativamente resultaba mucho mayor que cualquier cuerpo solar "normal". El resultado era que no se podía despreciar la presión de la luz radiada por el sol -el viento solar- como se hace con los cálculos de las órbitas "normales".[A partir de Weinberg, en (Klir, 1972, p.124)]["El viento solar" es un cuento de Arthur C. Clarke, Alianza Editorial, Colección el Libro de Bolsillo, Madrid, 1974].

Hoy, con toda la potencia de cálculo que los ordenadores ofrecen, es difícil comprender por qué los cálculos de las órbitas planetarias de Newton se cuentan entre los logros más importantes de la mente humana. Newton y sus contemporáneos estaban muy interesados por los supuestos simplificadores que les permitían realizar cálculos como el descrito. Con el límite práctico de cálculo muy por debajo del actual, Newton necesitaba de todos los supuestos simplificadores que pudiese encontrar y ahí está el secreto de su genialidad, seguir un proceso de simplificación que le permitió llegar a un problema abordable y que al mismo tiempo mantenía la validez de los resultados. También tuvo la suerte de que los instrumentos de observación de su época eran infinitamente menos precisos que los actuales.

El problema de la simplificación sigue siendo muy importante hoy en día. Establecer unos principios generales de comportamiento y unas líneas simplificadoras válidas de forma que los problemas complejos sean manejables no es una tarea sencilla. Cada problema requiere su método de simplificación particular y hay que ser muy cuidadoso con los factores que se desprecian, un ejemplo de los problemas que pueden resultar de no hacerlo así se resume en el cuadro "El viento del sol".

4. Tempus y Hora. Una estrategia de simplificación, según H.A. Simon

Dentro de su ensayo "La arquitectura de la complejidad", en el que trata diversos aspectos relacionados con la complejidad de los sistemas, Simon propone una de las parábolas emblemáticas del estudio de la complejidad, que ilustra perfectamente cómo se puede tratar ésta y los problemas derivados de ella. El cuento, o la parábola, como lo llama Simon, dice más o menos así:

Hubo una vez dos relojeros, llamados Tempus y Hora, que fabricaban relojes de excelente calidad. Los dos gozaban de una gran reputación y tenían un gran número de clientes. Sin embargo, Hora prosperó mientras que Tempus se fue empobreciendo cada vez más hasta que finalmente perdió el negocio. La razón de ello requiere una explicación previa.

Los relojes que ambos hacían estaban compuestos por unas 1.000 piezas. Tempus ensamblaba los relojes de tal manera que si interrumpía el proceso de fabricación en cualquier punto que no fuera el final de la construcción del reloj, éste se desmontaba y había que empezar el proceso de nuevo desde el principio. A medida que tenía más clientes que llegaban a su taller a formalizar sus encargos, sufría más interrupciones y se le hacía más difícil tener tiempo suficiente para acabar un reloj.

Los relojes que hacía Hora no eran menos complejos, pero estaban diseñados de tal forma que se podían construir partes completas por separado, cada una de ellas de unas diez piezas. Cada parte de éstas servía para construir otra subparte más grande de unas cien piezas y diez de estas subpartes formaban el reloj completo. De aquí que, cuando Hora se veía interrumpido por un cliente, con la misma frecuencia que Tempus, el trabajo que perdía era sólo una pequeña parte del trabajo total, consiguiendo montar un reloj en mucho menos tiempo que Tempus.

El propio Simon analiza numéricamente este ejemplo. Supongamos que la probabilidad de interrupción mientras se monta una pieza es p . La probabilidad que tiene Tempus de acabar un reloj es la probabilidad de que no le interrumpan mientras monta cada una de las 1.000 piezas, esto es, $(1 - p)^{1.000}$, que para p mayor que 0,001 es una probabilidad muy pequeña y supone que cada interrupción representa perder $1/p$ partes montadas. Por otro lado, Hora tiene que completar 111 montajes, cada uno de diez partes, de forma que la probabilidad de no ser interrumpido mientras completa cada una de estas subpartes es $(1 - p)^{10}$, aunque lo que más interesa es que cada interrupción supone que sólo se pierde el trabajo de montar 5 piezas.

Para $p = 0,01$ basta decir que Tempus conseguiría completar 44 relojes de cada millón de intentos mientras que Hora conseguiría acabar 9 de cada 10 subpartes de las que empieza, lo que se traduce en una relación de 4.000 a 1.

La moraleja evidente de esta parábola es que la complejidad se puede manejar muy eficientemente, aunque no en todos los casos, por medio de módulos o jerarquías, otro aspecto muy importante de la complejidad que trataremos aparte en un capítulo más extenso. El ejemplo que propone Simon es muy útil para ilustrar cómo reducir la complejidad descriptiva según el punto 3 que propone Klir.

5. Un paso más allá de la simplificación. Bell y Newell

Como decíamos al principio de este capítulo, la simplificación es una necesidad, dada la complejidad del mundo que nos rodea. Hemos visto diversos ejemplos de cómo se reduce la complejidad de un determinado sistema hasta tener un nivel asequible. Klir muestra los aspectos formales del problema de la simplificación, Weinberg lleva la idea aún más lejos considerando la simplificación como una ciencia ("La simplificación de la ciencia y la ciencia de la simplificación" [Weinberg, en Klir, 1978]) y nos proporciona un ejemplo muy ilustrativo de su importancia en la ciencia. Simon también propone un ejemplo para demostrar la importancia de las jerarquías y la modularidad en el tratamiento de sistemas complejos. Con todos estos autores hemos desarrollado una imagen muy completa de lo que es la simplificación, pero falta algo más.

La simplificación por sí sola no sirve para tratar sistemas complejos. Efectivamente es un problema muy importante pero es sólo un lado de la ecuación. La formulación completa de esta ecuación es la que aparece reflejada en la siguiente figura:

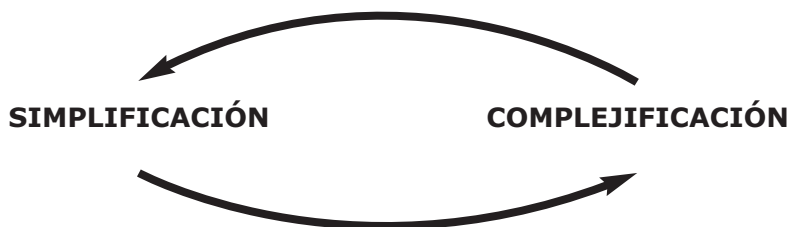


Fig. 1. La simplificación y la complejificación, como procesos complementarios

Este bucle define una jerarquía epistemológica, una espiral evolutiva del conocimiento [Morin, 1986, p. 63]. En un primer paso, al enfrentarnos con un sistema o con un objeto genérico hay que seleccionar algún método de simplificación que nos permita alcanzar el adecuado balance de variedades (ver el capítulo dedicado a **la variedad** y la ley de la variedad requerida), esto es el proceso que Weinberg quiere mostrar con el ejemplo de Newton. Una vez que se ha alcanzado un cierto grado de conocimiento sobre el sistema o el objeto, nuestra variedad se amplía y somos capaces de abarcar realidades más complejas pues disponemos de una teoría explicativa y de un lenguaje que nos permite nombrar lo que vemos (sobre la importancia del lenguaje volveremos más adelante). Entonces se produce el proceso complementario, complejificamos, esto es, desarrollamos a partir de ese modelo simplificado una serie de conocimientos que nos permiten aumentar el alcance y la profundidad de lo que sabemos.

Es un cambio permanente de perspectiva: primero, observamos la totalidad y la simplificamos para poder explicarla; cuando aprehendemos esa realidad se abren multitud de nuevas perspectivas -complejificación-, que de nuevo volvemos a simplificar en un nivel superior de conocimiento para poder profundizar en su estudio. La simplificación ilustrada por Weinberg conduce a la ley de gravitación universal y a una estructura determinada del sistema solar, una vez que ésta se conoce se puede aumentar el número de factores que consideramos y llegar a un nuevo modelo de sistema solar.

Este bucle se puede interpretar a la luz de los dos principios generales de la complejidad de Klir. La simplificación es una reducción de la complejidad descriptiva, la complejificación una disminución de la incertidumbre asociada con el sistema. Ambas son complementarias y constituyen las piezas fundamentales del ciclo cognitivo que ilustra la figura 1.

Para tratar este punto con más detalle, vamos a tomar como ejemplo la jerarquía que proponen Bell y Newell para un ordenador en la que se ilustra perfectamente cómo se debe de jugar tanto con la simplificación como con la complejificación para describir un sistema completo. Además, utilizar este ejemplo nos puede servir también como un primer nexo de unión con la parte de este trabajo dedicada a las tecnologías de la información y como ejemplo de aplicación de todas las ideas desarrolladas hasta ahora.

6. Niveles y Abstracciones

Ya hemos visto cómo se puede reducir la complejidad de un sistema si se organiza en varias partes o, lo que es lo mismo, en varios niveles. Esto nos permite entenderlo mucho mejor al separar diferentes propiedades y características en diferentes estratos que se pueden estudiar por separado.

Esto es lo que hicieron Bell y Newell con los ordenadores. Descomponiendo éstos en varios niveles se facilita el estudio de sus diversas funcionalidades y el análisis de sus características. Para ello se manejan dos conceptos fundamentales: abstracción y refinamiento. El paso de un nivel inferior de la jerarquía a otro superior se hace por abstracción, es decir, suprimiendo detalles innecesarios y quedándose sólo con la información relevante en cada nivel. El paso de un nivel superior a otro inferior se hace por refinamiento, es decir, ampliando los detalles de ese nivel.

Dicho de otra manera, cada nivel del sistema se caracteriza por una serie de componentes y una serie de formas de relacionarlos para formar estructuras. El comportamiento de un nivel se define en términos del comportamiento de sus componentes y de las combinaciones específicas. Para pasar a un nivel superior se abstraen los detalles de los componentes y nos quedamos sólo con lo relevante a ese nivel. Para descender a un nivel inferior se amplían los detalles de esos componentes, se refina la estructura con un grado de discriminación cada vez mayor.

Y ésta es una característica recursiva: un sistema, formado por una serie de componentes organizados de una determinada manera, puede verse como un componente de un sistema superior. En todos los niveles se pueden distinguir sistemas que no se pueden explicar a partir de estructuras de sistemas del mismo tipo, estos sistemas se denominan primitivas. Una resistencia, por ejemplo, es una primitiva en un circuito, pero no siempre hay primitivas puras (los operadores lógicos AND, OR, NOT, NAND, NOR, etc. pueden todos representarse unos en función de otros, luego no se podría hablar de primitivas estrictamente).

El ordenador, como jerarquía de niveles

Siewiorek, Bell y Newell proponen una interpretación jerárquica del ordenador en la que los niveles inferiores son "los ladrillos" con los que se construye el nivel inmediatamente superior. En cada nivel se pueden distinguir componentes, que se pueden ver como primitivas en ese nivel, y estructuras construidas a partir de varios componentes.

El nivel más bajo sería el nivel de circuitos, los componentes en este nivel son resistencias, condensadores, bobinas, transistores, etc. Las estructuras que con ellos se construyen son los amplificadores, retardos, relojes, puertas lógicas, etc.

El siguiente nivel sería el nivel de diseño lógico, a su vez dividido en los subniveles de transferencia de registros y de conmutación de circuitos. En este último los componentes son las puertas, flip-flops, "latches", retardos, etc. Las estructuras son contadores, registros, codificadores y decodificadores. En el subnivel de transferencia de registros los componentes son los registros, contadores y demás, mientras que las estructuras ya son unidades aritméticas, secuenciadores. Como estructuras superiores de este nivel estarían los microprogramas y las microrrutinas.

Estos dos niveles son suficientes para ilustrar cómo cada nivel tiene unos componentes básicos y unas estructuras que se construyen a partir de ellos y también cómo cada nivel tiene su lenguaje propio que sólo se entiende en ese nivel. Para los circuitos una parte de ese lenguaje es la ley de Ohm, que no tiene ningún sentido en el nivel de diseño lógico, donde el lenguaje es el álgebra de Boole, por ejemplo. La existencia de estos lenguajes diferenciados es una prueba de la existencia de estos niveles.

Esta jerarquía considera también el nivel de programa, dividido en varios subniveles, donde se utilizan los diferentes lenguajes de programación que existen, lenguaje ensamblador para el subnivel más bajo, primitivas del sistema operativo, lenguajes de programación de aplicaciones, módulos de aplicaciones (librerías de funciones matemáticas, por ejemplo) y aplicaciones completas (como puede ser un sistema de reserva de billetes). Por encima de este nivel está el de Procesadores-Memoria-Conmutadores, en el que se manejan redes de ordenadores, sistemas multicomputador, etc.

Cada sistema se constituye así en un nivel de la jerarquía en que se descompone el sistema total y se caracteriza por un lenguaje distintivo que permite representar los componentes de ese nivel (componentes, leyes de comportamiento y relaciones) y además representa propiedades especiales de esos componentes pues de no ser así no tendría sentido utilizar un lenguaje específico. El hecho de que estos lenguajes son muy distintos aumenta la confianza sobre la existencia de diferentes niveles. Esto se puede relacionar directamente con la jerarquía de niveles que propone Klir,

en la que cada nivel epistemológico tiene un lenguaje concreto y distintivo asociado. Este proceso se puede ver en la figura 3. La jerarquía que proponen Bell y Newell se resume en la figura 2. [Siewiorek, Bell y Newell, 1982, p. 10].

PMS level			Structures: Networks, multiple processor systems, computers
			Components: Processors, memories, switches, controllers, transducers, data operators, links
Program level	High level language sublevel	Application systems	Structures: Statistical Package for the Social Sciences (SPSS), partial differential equation solver, power system simulator, airline reservation system
		Application routines	Components: Mathematical library routines, formatting routines
		Run-time system	Structures: Mathematical functions, plotting packages
		Operating system	Components: Subroutines, memory allocation
			Structures: Memory allocation, input/output, file system
			Components: Operating system calls
			Structures: Schedulers, allocators, communication
			Components: Subroutines, coroutines, programs
	ISP sublevel		Structures: Instruction sets
			Components: Memory state, processor state, effective address calculation, instruction decode, instruction execution
Logic design level	Register transfer sublevel	Control	Structures: Microprograms, microroutines
		Hardwired	Components: Microsequencer, Microstore
		Data paths	Structures: Sequencer
			Components: Sequential machines
	Switching circuit sublevel	Sequential	Structures: Arithmetic units, register files
Combinational		Components: Registers, data operators	
			Structures: Counters, functional generators, registers
			Components: Flip-flops, latches, delays
			Structures: Encoders, decoders, data operators
			Components: Gates
Circuit level			Structures: Amplifiers, delays, clocks, gates
			Components: Transistors, relays, resistors, capacitors

Fig.2. Jerarquía descriptiva de los ordenadores (Bell y Newell, 1982)



Fig. 3. Jerarquía de niveles, cada uno con su lenguaje correspondiente

7. Resumen

La simplificación es un concepto indisoluble de la complejidad. Como hemos señalado repetidas veces a lo largo del capítulo, la simplificación es una necesidad que viene impuesta por la complejidad de las cosas. Varios ejemplos nos han servido para ilustrar cómo se aplican diferentes formas de simplificación a diversos sistemas para conseguir reducirlos a una complejidad manejable. Pero más que el proceso en sí, nos interesa destacar la relación que existe entre simplificación y muchos conceptos relacionados con la complejidad que hemos tratado en capítulos anteriores.

Los principios generales de complejidad de Klir nos proporcionan la primera pista al incluir la incertidumbre en la complejidad de los sistemas. El bucle simplificación/complejificación nos permite relacionar la disminución de la complejidad descriptiva con la simplificación y, por tanto, con el aumento de incertidumbre. De ahí podemos saltar al Azar, pues la pregunta de si existe un Azar en sí mismo o es producto de nuestra ignorancia está directamente relacionada con el binomio descripción/incertidumbre. Si admitimos que no podemos describir las cosas en su totalidad (como postula el principio de Heisenberg) siempre existirá una incertidumbre asociada, incertidumbre que se traduce en Azar (ver el capítulo dedicado a los **conceptos relacionados con la complejidad**). ¿Hasta dónde se puede llegar siguiendo el bucle epistemológico de complejificación/simplificación?; ésta es una pregunta que queda abierta y que invitamos al lector a responder desde la perspectiva global de este trabajo.

Por otro lado, la simplificación puede intentar formalizarse a través de la Ley de Variedad Requerida formulada por Ashby y que recogemos en el capítulo dedicado a la **Variedad**. La simplificación está encaminada a equilibrar las variedades, actuando como amplificador y reductor de variedad a un tiempo, y ha de cumplir

las leyes cibernéticas sintetizadas en esta ley. También invitamos al lector a reflexionar sobre esta formulación.

8. Bibliografía

Dividida en dos apartados. En el apartado Notas Bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

El problema de la simplificación ha sido tratado por muchos autores y la bibliografía disponible es muy abundante, aunque un tanto dispersa al encontrarse las referencias a este tema dentro de obras con unos objetivos más amplios. Los títulos que aquí hemos utilizado como referencia son los siguientes:

Gerald M. Weinberg, "Una aproximación por computadores a la Teoría General de Sistemas", artículo que se encuentra en el libro editado por George J. Klir "**Tendencias de la Teoría General de Sistemas**", de la colección Alianza Universidad Ciencias, num. 208, Madrid 1978. Este artículo puede resultar muy interesante a los aficionados a la informática pues presenta muchas de las ideas sobre complejidad y sistemas a través de ejemplos donde el ordenador juega un papel primordial. El libro, editado por Klir, es una obra importante, que ya hemos utilizado varias veces en otros capítulos, y que recoge diversos artículos sobre sistémica de varios reconocidos expertos del tema (Ashby, Bertalanffy, Weinberg, el propio Klir, etc.) y que puede proporcionar una panorámica muy amplia de lo que es el enfoque sistémico, además tiene la ventaja de estar en español.

George J. Klir, "Complexity, Some General Observations", **Systems Research**, No. 2, pp. 131-140, 1985. Artículo básico en el estudio de la complejidad y que recoge gran parte de las ideas de su autor. Este artículo también nos ha servido de referencia en el capítulo de Marcos Conceptuales y la mayoría de las ideas que en él se presentan aparecen también en el libro de Klir que se comenta en el capítulo de Conceptos relacionados con la Complejidad.

Herbert A. Simon, "The Architecture of Complexity", ensayo recogido en el libro **The Sciences of the Artificial**, MIT Press, 1969. Éste es un pequeño libro con una serie de conferencias de Simon que tocan puntos muy diversos y muy interesantes. El artículo que tomamos como referencia es muy conocido y frecuentemente citado, tanto por los partidarios del enfoque sistémico como por sus detractores, dado que es una aproximación imparcial a los sistemas con críticas y propuestas de desarrollo muy acertadas. El resto del libro es también muy interesante y recomendable.

Siewiorek, Bell y Newell, **Computer Structures: Principles and Examples**, McGraw Hill, Nueva York, 1982. Es una auténtica biblia y enciclopedia de la estructura y arquitectura de los computadores desde su primera aparición, sin Siewiorek, en 1971. Utiliza en sus primeros capítulos un método descriptivo jerárquico y define los lenguajes ips y pms.

Referencias bibliográficas

Klir G.J. (1985) "Complexity, Some General Observations", **Systems Research**, No. 2, pp. 131-140 .

Siewiorek, Bell y Newell (1982), **Computer Structures: Principles and Examples**, McGraw Hill, Nueva York.

Simon H.A., (1969) "The Architecture of Complexity", ensayo recogido en el libro **The Sciences of the Artificial**, MIT Press.

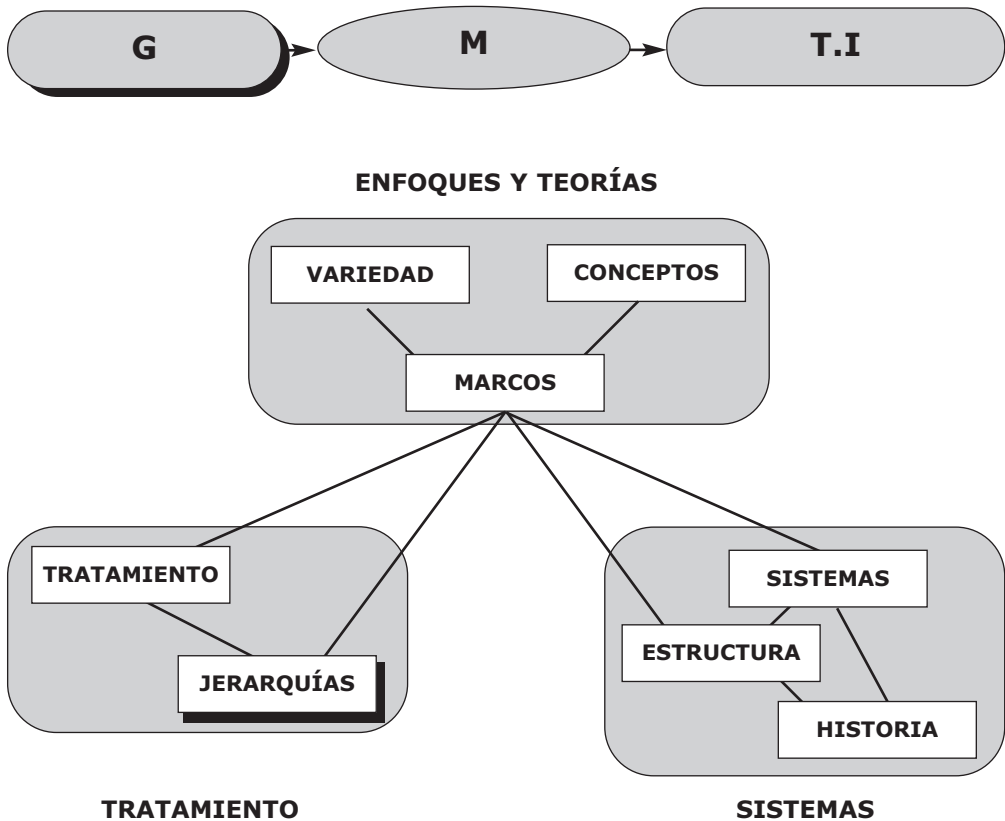
Weinberg G.M. (1978), "Una aproximación por computadores a la Teoría General de Sistemas", en George J. Klir, **Tendencias de la Teoría General de Sistemas**, de la colección Alianza Universidad Ciencias, num. 208, Madrid.

Capítulo 8: Las jerarquías

Los sistemas que plantean problemas de gran complejidad no se prestan a una descripción que a la vez sea suficientemente detallada y comprensible en su totalidad por un observador. La aproximación jerárquica nos permite descomponer el sistema complejo en niveles, cada uno de una complejidad menor, y nos da una estructura para entender el conjunto.

Pero no sólo es útil el concepto de jerarquía a la hora de describir un sistema. También es una herramienta adecuada para ayudar a la toma de decisiones, o a la organización. Esto es debido, entre otras cosas, a las ventajas que aporta en la construcción o comprensión de la complejidad, y en el proceso de información.

La jerarquía, como herramienta, puede considerarse como un potente ejemplo de filtro de variedad, cuyo uso, además, está muy extendido en las ciencias.



1. Introducción

Podemos decir que, "casi por definición, los sistemas verdaderamente complejos escapan a una descripción completa y detallada" [Mesarovic y Macko, 1969]. El problema es que si queremos estudiar el sistema con gran detalle, llegamos a una situación donde "los árboles no nos dejan ver el bosque", donde perdemos por completo la visión de conjunto, y con ella la comprensión del sistema como un todo. Por otra parte, si intentamos abordarlo desde el punto de vista más general, buscando un modelo "sencillo", que sea comprensible, hemos de dejar de lado los detalles, lo que en un sistema complejo es frecuentemente inadmisibile.

Para sortear este callejón aparentemente sin salida, echaremos mano del concepto de jerarquía. Pero no entendiéndolo como un conjunto parcialmente ordenado por una relación de dominación, sino en el sentido más amplio de división de los problemas en niveles de organización [Bunge, 1969]. De esta forma, en cada nivel nos dedicaremos a estudiar todos sus componentes con el detalle necesario, pero con la ventaja de que podemos apoyarnos para ello en una visión "sencilla" del nivel inferior. Así tenemos a la vez el detalle y la visión de conjunto, lo que nos permitirá manejar con mayor facilidad la complejidad del sistema.

Comenzaremos por analizar los aspectos comunes a las distintas formas de entender el concepto de jerarquía. Aparecerá así la idea de estrato, central a la hora de tratar con jerarquías. Simon nos proporcionará la primera definición, basada en la idea de "subsistemas interrelacionados en cada nivel".

Más adelante, Whyte introduce las jerarquías estructurales, concepto clave a la hora de entender cómo se enfrenta la ciencia a los problemas que trata de resolver. Mesarovic y Macko, por su parte, diferencian tres conceptos que aclaran un poco más nuestra idea de jerarquía, según esta idea se aplique a la descripción, a la toma de decisiones o a la organización.

Veremos también una formalización matemática del concepto de jerarquía, para aportar un poco de rigor a las definiciones vistas hasta ese momento. Y se pondrán una ideas (debidas a Wells) sobre cómo se manifiestan las jerarquías en el mundo real. Por último, realizaremos un pequeño estudio que pretende responder a la pregunta de por qué la jerarquía es un modelo de organización tan extendido.

2. Una aproximación al concepto de jerarquía

Para entrar en contacto con la idea de jerarquía de una forma "suave", recurriremos a las opiniones de un filósofo [Grene, 1969]. Dado que la intención es tratar el tema desde un punto de vista lo más científico posible, esta actitud puede parecer extraña. Pero tiene la ventaja de que estas opiniones enlazan muy bien con las ideas intuitivas que cualquier persona suele tener sobre las jerarquías.

Cuando estudiamos los conceptos en que se basan las diferentes ramas del saber, podemos apreciar que el término "jerarquía" aparece con mucha frecuencia. Pero en una primera aproximación da la impresión de que cada uno lo usa con un significado distinto. Así, oímos a los astrofísicos, y nos quedamos con la idea de clasificación por niveles, de acuerdo al tamaño (o a veces, a la densidad media) de los cuerpos. Los biólogos, sin embargo, no parecen preocupados por este tipo de cuestiones a la hora de establecer sus clasificaciones. Ellos se centran en la búsqueda de una organización donde la disposición de los elementos de un nivel restrinja el comportamiento de esos elementos, favoreciendo la emergencia de nuevas propiedades. Aparece de esta manera el concepto de regulación: la distinción de un nuevo nivel tendrá lugar al manifestarse un nuevo sistema que ejerce algún tipo de control sobre los elementos que lo componen (que serán por su parte miembros del nivel inferior).

Y de esta manera podríamos seguir explorando diversos campos científicos, cada uno con su propio concepto de jerarquía. ¿Es posible, después del panorama con que nos hemos encontrado, descubrir características comunes en todas las interpretaciones de esta palabra?. Si las estudiamos cuidadosamente, veremos que sí: en todas ellas hallamos la idea de una **clasificación en estratos**. Además, esta clasificación se hace según el **rango** de algún conjunto de elementos que caracterizan a los objetos de los que se ocupa. Como acompañante, puede aparecer la idea de **regulación** de unos estratos sobre otros, pero ésta no es esencial para que se hable de jerarquía (por ejemplo, en el caso de los astrofísicos, que consideramos antes, no era relevante).

NIVEL-4: Corpúsculos (partículas fundamentales).

NIVEL-3: Átomos.

NIVEL-2: Moléculas.

NIVEL-1: Sistemas moleculares.

a. Cristales.

b. Coloides.

NIVEL 0: Agregados coloidales y cristalinos.

a. Inorgánicos (minerales, meteoritos,...).

b. Orgánicos (organismos, colonias,...).

NIVEL+1: Asociaciones meteoríticas.

a. Chaparrones de meteoros.

b. Cometas.

c. Nebulosas coherentes.

NIVEL+2: Sistemas de satélites.

NIVEL+3: Estrellas y familias de estrellas.

NIVEL+4: Cúmulos estelares.

NIVEL+5: Galaxias.

a. Brillantes.

b. Tenues.

NIVEL+6: Agregaciones de galaxias.

NIVEL+7: Metagalaxia.

a. Cuerpos y sistemas siderales organizados.

b. Cosmoplasma o matriz.

NIVEL+8: El universo (complejo espaciotemporal).

Fig. 1. Ejemplo de clasificación jerárquica: resumen de la clasificación de los sistemas materiales, debida a Shapley [H. Shapley, 1958]

Así, en la Astrofísica, el rango en función del cual se realiza la clasificación viene dado por el tamaño de los objetos, mientras que para los biólogos lo determinante es el tipo de propiedades que tienen los elementos de cada estrato. El hecho de que una célula esté en un nivel más bajo que un organismo pluricelular se debe básicamente a las nuevas propiedades que emergen en este último.

Con esto hemos introducido ya la mayor parte de los conceptos que aparecen en el estudio de las jerarquías: clasificación, estratos, regulación (o control) y emergencia. En los siguientes apartados profundizaremos más en su significado e implicaciones.

3. La primera definición

Simon [Simon, 1962] estudia la jerarquía y sus relaciones con la complejidad. A él se debe la primera definición de jerarquía que vamos a presentar:

"Un sistema jerárquico es el que está formado por **subsistemas interrelacionados**, donde estos subsistemas son de tal forma que a su vez tienen una estructura jerárquica".

Como puede observarse, es ésta una definición recurrente. Se aplica comenzando por el sistema más global, y se va descendiendo con ella por sistemas cada vez más sencillos hasta llegar a lo que podríamos llamar subsistemas elementales. La estructura jerárquica así descrita lleva implícito el concepto de nivel, que estará constituido por todos los subsistemas interrelacionados que son componentes de uno dado (que estaría en el nivel inmediatamente superior). Sin embargo, es importante resaltar que no incluye la idea de subordinación, algo que suele ir acompañando a la idea de jerarquía en su acepción más usual. Así tienen estructura jerárquica tanto un ejército como una célula.

A partir de su definición, Simon introduce el concepto de **apertura** que, para un sistema dado, es el número de subsistemas que lo forman. Apoyándose en él diferencia dos tipos de jerarquías:

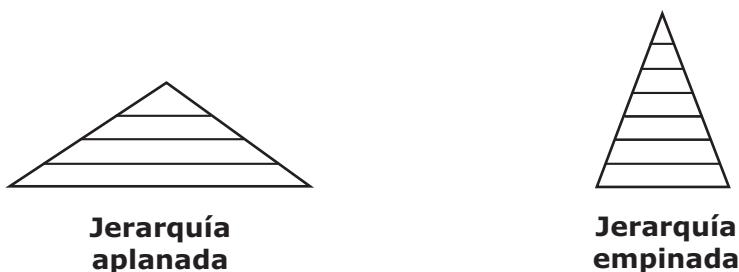


Fig. 2. Representación de una estructura jerárquica aplanada y otra, empinada (Simon).

- a. **Aplanadas:** serán aquéllas en las que la relación entre el número de niveles y la apertura sea pequeña (algo como $1:10^{10}$, en un volumen de gas molecular o en un diamante, por ejemplo). Podrían representarse mediante una pirámide de poca altura y gran base.
- b. **Empinadas:** La relación puede ser ahora mucho mayor (algo como $1:10$). Y la pirámide con la que pueden relacionarse tendrá una base pequeña, pero mucha altura.

4. Las jerarquías estructurales en la ciencia

Whyte, en su artículo [Whyte, 1969], propone un concepto que, si bien más restringido que el de Simon, incluye algunos aspectos interesantes.

Supongamos que las entidades de que se ocupa la ciencia se encuentran dispuestas formando secuencias ordenadas de niveles (en el sentido de que podemos hablar de "nivel superior" o de "nivel inferior") distintos y separadamente filiables. Diremos que estos elementos forman **jerarquías estructurales**.

Parece ser que en este tipo de estructuras la existencia de una "unidad" en un nivel dado impone ciertas restricciones sobre los grados de libertad de las partes que la forman (que son elementos del nivel inmediatamente inferior al considerado). Para resaltar este hecho se introduce la idea de **unidad estructural**, que será la unidad que impone ciertas restricciones características sobre el margen de libertad de las partes que la forman. Se da el caso, además, de que las propiedades de la unidad no se explican cómo una simple adición de las propiedades de las partes, sino que más bien parecen emerger por encima de éstas. Esta **emergencia** puede producirse o bien por superación de un cierto umbral global en alguna propiedad, al sumarse las de las partes (síntesis de una nueva forma, efecto de aglomeración,...), o bien por superarse umbrales locales en una unidad (fragmentación).

Por ejemplo, una célula, considerada como un sistema compuesto por distintos tipos de moléculas, sería una unidad estructural. Las partes que la forman serán las moléculas que entran a formar parte de su composición. En este caso parece claro que el hecho de pertenecer a la célula impone algunas restricciones sobre los grados de libertad de dichas moléculas. Ahora no podrán interactuar de cualquier manera, porque las condiciones energéticas y ambientales están controladas de forma que la célula mantenga su vida. Por otra parte, es sencillo encontrar propiedades emergentes en el sistema total que no aparecen en las partes. Esto se refleja incluso en la ciencia que se ocupa de cada nivel: al pasar de moléculas a células pasamos de hablar en términos químicos a hacerlo en términos biológicos.

En el caso de la célula se dan las dos formas de emergencia de nuevas propiedades antes mencionadas. Por un lado podemos citar como caso de aglomeración la acumulación de algunos tipos de moléculas en la periferia, formando la membrana, de gran importancia para mantener el adecuado equilibrio en las relaciones con el medio ambiente. Y como ejemplo de emergencia de propiedades por fragmentación tenemos las distintas concentraciones locales de moléculas en el citoplasma de la célula, que llevan a la aparición de los diferentes orgánulos celulares: mitocondrias, ribosomas, vacuolas,...

Como ya habíamos adelantado, el concepto de jerarquía que hemos perfilado aquí es más restringido: el paso de un nivel a otro debe ahora implicar cierta "limitación de libertad" del nivel inferior. Se introduce así la idea de control de unos estratos sobre otros. Y se da la **emergencia** de propiedades nuevas al pasar de un nivel a otro superior como característica fundamental de las estructuras jerárquicas.

5. Tres nociones distintas, pero relacionadas

Al hablar de jerarquías, van apareciendo varias ideas que, si bien son parecidas, no son exactamente iguales. Para clarificar el tema, Mesarovic y Macko [Mesarovic y Macko, 1969] proponen una diferenciación entre tres conceptos:

1. **Jerarquía de estratos.**
2. **Jerarquía multicapa de toma de decisiones.**
3. **Jerarquía de organización.**

5.1. Jerarquía de estratos

Cuando nos enfrentamos a un sistema verdaderamente complejo, nos es prácticamente imposible "encerrarlo" en una descripción que sea a la vez detallada y fácilmente comprensible. Normalmente nos encontramos con el dilema de decidir entre sencillez y detalle. La sencillez es necesaria para que la descripción sea de alguna utilidad (si es demasiado complicada, simplemente no entenderemos nada). El detalle, por su parte, es preciso si queremos tener en cuenta todos los aspectos del sistema de la forma más realista posible.

Pues bien, desde hace mucho tiempo se utiliza, de una forma más o menos consciente, una técnica que nos permite soslayar este problema. Podemos realizar una descripción por estratos del sistema o, dicho de otra forma, una descripción jerárquica. Para ello utilizaremos una familia de modelos, cada uno de los cuales se ocupará del sistema desde un nivel de abstracción distinto. De esta forma, tendremos las variables, relaciones, principios, etc., de cada estrato de nuestra descripción con el detalle que consideremos oportuno.

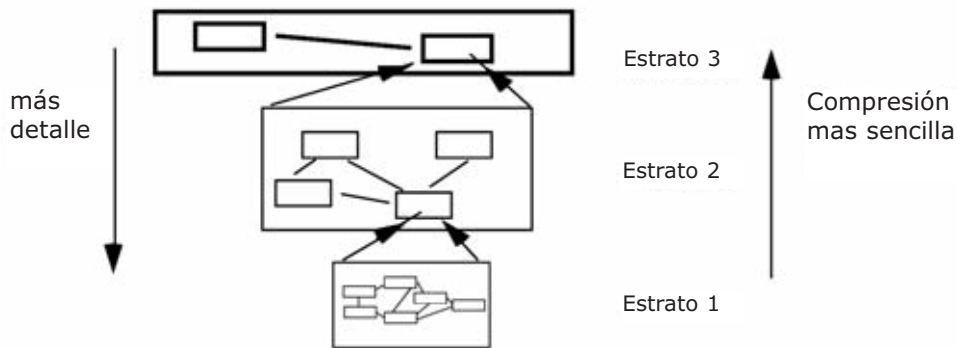


Fig. 3. Esquema de un sistema estratificado

Pero no basta con esto para que nuestra descripción cumpla además el objetivo de la sencillez. Necesitaremos también que cada uno de los niveles de la descripción pueda considerarse independiente de los otros, en el sentido de que fije sus propias relaciones y características sin necesidad de hacer referencia expresa a otros niveles. A un sistema descrito de esta manera lo llamaremos **sistema estratificado**.

En una descripción de este tipo, podemos situarnos en un nivel dado, y "explorarlo", con relativa independencia de los demás. Esto nos permite llegar a conclusiones, y a un grado de comprensión que no podríamos alcanzar si intentásemos abarcar el sistema de una forma global.

Veamos algunos ejemplos para aclarar lo dicho hasta aquí. Para empezar, buscaremos en las ciencias naturales. Así nos encontramos con la estratificación del estudio de los seres vivos. Los niveles utilizados en este caso son: molecular, celular, de órganos y de organismos. Desde luego, en la realidad estos niveles no están totalmente desligados unos de otros. Pero el suponerlo a la hora de analizarlos nos permite una mayor comodidad y facilidad de profundización en cada uno de los estratos considerado por separado. En cualquier caso, no debemos perder de vista que las interrelaciones entre niveles están ahí. Si no las tenemos en cuenta sólo conseguiremos una percepción incompleta del comportamiento de sistema global. De hecho, al centrarnos en un enfoque puramente biológico de los seres vivos ya nos estamos aislando de otros puntos de vista igualmente posibles (a otros niveles de abstracción), como podrían ser el físico-químico o el ecológico, sin los cuales no podremos explicar determinadas características del sistema (en nuestro caso, los seres vivos).

Otro ejemplo en el que estamos acostumbrados a un estudio jerárquico es el del ordenador. Es normal pensar en él como, al menos, el conjunto formado por dos niveles: el puramente electrónico y el informático. En el nivel electrónico tenemos todos los circuitos que componen el ordenador, y las relaciones entre ellos. Las variables a este nivel serán intensidades de corriente eléctrica, tensiones, etc. Los elementos básicos son los transistores, resistencias, condensadores... En el nivel informático, sin embargo, el ordenador puede ser considerado como un procesador de información. Hablaremos ahora de flujos de datos y de operaciones lógicas, de cantidades de información, etc. En cada uno de los dos niveles ignoramos casi totalmente la existencia del otro, lo que nos permite comprender el sistema de una forma mucho más sencilla, aunque incompleta. Como pequeño ejercicio puede imaginarse, por un momento, lo que supondría seguir el proceso de ejecución de un programa atendiendo únicamente a las intensidades de corriente que pasan por los transistores de la CPU. Por otra parte, los dos niveles de los que hemos hablado pueden a su vez ser subdivididos en otros muchos (habitualmente lo son). Así, para el informático, se definen las "máquinas virtuales" a distintos niveles de abstracción: la que nos proporciona el lenguaje máquina, los de alto nivel, o alguna interfaz de usuario.

De los ejemplos dados (y de alguno más que pueda descubrir el propio lector), pueden extraerse algunas características generales de la descripción estratificada de sistemas:

- a. El **observador** es quien elige los estratos que describen el sistema, según sus conocimientos, sus intereses, etc. Un observador interesado sólo en la electrónica, podrá teóricamente dar una descripción completa desde el punto de vista físico de un ordenador, con todos sus circuitos detallados. Pero si no tiene noción de ello, no puede describirlo como procesador de información: para ello tendría que pasar a otro nivel, del nivel físico al informático.
- b. En general, los niveles son bastante independientes, en el sentido de que las leyes que gobiernan el comportamiento del sistema en un estrato dado no pueden deducirse de los principios empleados en otros. Por ejemplo, a partir de las leyes de la física no pueden deducirse las de la informática. Podemos hablar de **independencia de comprensión**.
- c. Hay una interdependencia asimétrica entre los distintos estratos (**interdependencia funcional**). Con esto nos referimos a que para el correcto funcionamiento de un nivel es preciso que todos los que se encuentran por debajo de él también funcionen adecuadamente. Así, los requisitos que deben verificarse en un estrato para que cumpla adecuadamente su funcionalidad suelen encontrarse en los estratos inferiores como constricciones impuestas a sus comportamientos. Veamos un ejemplo: Para que un ordenador funcione adecuadamente, es necesario que los componentes electrónicos de que está

formado operen dentro de unas ciertas condiciones (así, los transistores sólo estarán en cortocircuito o en saturación, para darnos los dos niveles digitales).

- d. Cada estrato tiene su propio lenguaje, sus propios conceptos y principios. Y en cada estrato el sistema y sus objetos son definidos de una forma distinta. Normalmente, a medida que subimos en la jerarquía de estratos para un sistema cualquiera, vamos perdiendo detalle en la descripción. Es como si al ir alejándonos del sistema fuésemos percibiéndolo más desdibujado, y sin los detalles que podíamos apreciar cuando estábamos más cerca. De esta forma, lo que en un estrato es considerado un sistema, en el inferior se descompone en un conjunto de subsistemas. En el caso del ordenador, en el nivel electrónico tenemos transistores, resistencias, condensadores... Pero subiendo un nivel, nos los encontramos agrupados en circuitos (memorias, sumadores, amplificadores). Y aún más arriba, en equipos (ordenadores). En los estratos inferiores nos dedicamos al estudio por separado de los subsistemas, mientras que en los superiores atendemos especialmente a las relaciones entre éstos (ver el capítulo "Tratamiento de la complejidad").
- e. Cuando descendemos en la jerarquía de estratos, obtenemos más detalle, pero cuando ascendemos, profundizamos más en la comprensión del sistema. Así, el biólogo se basa en la duplicación de las cadenas de ADN para explicar los fenómenos de la herencia. El bioquímico se basa en las propiedades de los nucleótidos para explicar ésta duplicación. El químico explica la formación de parejas de nucleótidos basándose en la formación de enlaces de hidrógeno. A su vez, el físico molecular explica éstos utilizando funciones del potencial intermolecular, que el físico cuántico se encarga de estudiar a partir de la ecuación de ondas...

5.2. Jerarquía multicapa de toma de decisiones

Otro concepto asociado con las jerarquías aparece cuando estudiamos los procesos de toma de decisiones. El dilema viene dado ahora por la necesidad de tomar la decisión cuanto antes (limitación temporal de actuación), pero tomándonos el tiempo suficiente para poder comprender la situación y decidir de una forma acertada. Para poder conjugar adecuadamente estas dos aspiraciones es frecuente una aproximación jerárquica: dividimos la decisión en una secuencia de problemas decisorios más sencillos (capas decisorias), y los vamos resolviendo uno a uno. Cada vez que resolvemos uno de estos "subproblemas", fijamos una serie de condiciones sobre el siguiente, que harán más fácil su resolución, y así sucesivamente.

Tenemos así lo que podemos denominar **jerarquía de capas decisorias**, construida de tal forma que la resolución de todas las capas implica la resolución del problema original. El sistema de toma de decisiones que funciona de esta forma se llama **sistema de capas múltiples de decisión**.

Veamos un ejemplo de la vida cotidiana que nos puede ilustrar estos conceptos. Podríamos pensar que la meta de una persona dada es ser feliz. Pero, así especificada, es ésta una meta muy poco definida: con gran probabilidad la concretará, según sus creencias, en estudiar una cierta carrera, buscar un tipo de trabajo, o elegir una determinada forma de vida. Una vez elegido este objetivo concreto, vendrá la toma de decisiones "encadenadas". Si nuestro personaje elige ir a la universidad, tendrá que comenzar por decidir la carrera en la que se matricula. Luego, la especialidad que cursará, si seguirá hasta doctorarse o no, etc. Y normalmente sólo una vez que hemos alcanzado una de esta submetas (por ejemplo, acabar la especialización), podremos evaluar si nos estamos acercando o no a la meta de ser feliz, y basándonos en esa evaluación tomaremos la siguiente decisión. Además, cada decisión que ponemos en práctica restringe el número de posibilidades que tenemos al tomar la siguiente (si elegimos la carrera de medicina, a la hora de buscar trabajo no tendremos que preocuparnos por la posibilidad de ejercer como abogado). De esta forma la toma de decisiones se ve facilitada, ya que nuestra mente sólo puede tener en cuenta un número relativamente pequeño de opciones.

5.3. Jerarquía de organización

Este concepto aparece en sistemas compuestos por un conjunto de subsistemas interaccionantes, alguno de los cuales son unidades de toma de decisiones. Dichas unidades han de estar organizadas de forma que haya una jerarquía establecida sobre ellas. Esto es, es preciso que unas manden sobre otras (en el sentido de que influyan en sus decisiones). Llamaremos a estos sistemas **de metas y niveles múltiples**, por la gran importancia que tiene en ellos el que las unidades de decisión tengan metas en conflicto. Esta multiplicidad de metas será necesaria para que el sistema, en su totalidad, funcione de forma adecuada.

En las organizaciones humanas tenemos un destacado ejemplo de estas jerarquías. En ellas cada individuo representa un centro de decisión, condicionado por las estructuras de mando que actúan sobre él. En principio, tiene sus propios objetivos, aunque inevitablemente la jerarquía que impone la organización influye fuertemente en las decisiones que puede tomar.

Los siguientes factores parecen explicar el gran éxito de esta forma de organización:

- a. Dado un conjunto de subsistemas independientes pero interactuantes, permite integrarlos en un sistema global de una forma bastante natural.
- b. Permite abordar tareas que sobrepasan, muchas veces con creces, la capacidad (física, pero también de toma de decisiones) de las unidades.
- c. Los recursos disponibles pueden ser utilizados de una forma muy eficiente.

- d. Las perturbaciones locales se propagan con gran dificultad a otras partes del sistema. Esto proporciona una gran fiabilidad. Pero también puede llevar a importantes inercias.

Una de las principales características de los sistemas de metas y niveles múltiples es que, aunque las unidades de rango jerárquico inferior pueden estar muy condicionadas por las de rango superior, siempre se mueven dentro de un margen de libertad en lo que se refiere a su toma de decisiones. Puede mostrarse que esa "libertad limitada" es esencial para el funcionamiento eficiente de estas estructuras. Y esto cuenta no solo para los sistemas "naturales" (sociales, biológicos,...) sino también para los construidos por el hombre.

6. Una formalización matemática

Bunge [Bunge, 1969] propone una formalización matemática del concepto de jerarquía que engloba bastante bien todas las aportaciones estudiadas.

De una forma estricta, puede entenderse el concepto de jerarquía como una relación de subordinación (y dominación) que afecta a un conjunto. Basándonos en esta idea, pasamos a la siguiente definición formal:

"Sean C un conjunto no vacío, i un elemento de C , y D una relación binaria en C . Decimos que la terna ordenada $J = \langle C, i, D \rangle$ es una **jerarquía** si, y sólo si, se cumple que:

- a. i es el elemento iniciador de C .
- b. i se encuentre siempre "por encima" de cualquier otro elemento de C , en el sentido marcado por la relación D .
- c. Para cualquier elemento y de C (excepto i), hay un único elemento x de C tal que Dxy (esto es, x "domina" a y).
- d. La relación D es antisimétrica y transitiva.
- e. La relación D representa dominación o poder.

D es por tanto una relación de dominio. Desde este punto de vista, podemos interpretar que a. y b. fijan al elemento i como "jefe supremo" de la estructura. La condición c. supone que cada elemento tenga un jefe y sólo uno. El flujo de poder, y el sentido de éste de "arriba abajo" viene dado por d.

De esta definición puede deducirse de forma natural el concepto de rango:

"Decimos que un subconjunto C_n de C , que no sea i , es el n -ésimo rango de J si, y sólo si, para todo x de C_n , x se encuentra n escalones "por debajo" de i (según la relación D). Se entiende escalón como una única aplicación de la relación D "

Así pues, podemos representar la jerarquía que hemos definido como un árbol finito. No podrá contener lazos cerrados, y se ramificará a partir de un solo punto (que será el que se corresponda con el elemento i). Tenemos, por tanto, lo que en teoría de grafos se llama **grafo orientado abierto**. Sin embargo es importante hacer notar que la condición e. de las que antes impusimos es puramente semántica, de significado de la relación, y por tanto no representable mediante el grafo.

Una vez definido lo que entendemos por jerarquía, pasemos a ver qué es lo que consideramos estructura de niveles:

"Sea un par ordenado $N = \langle C, E \rangle$, donde C es una familia de conjuntos de sistemas individuales, y E una relación binaria en C . Decimos que N es una **estructura de niveles** si, y sólo si:

- a. Todo conjunto miembro de C es una clase de equivalencia de sistemas (esto es, los sistemas de ese conjunto comparten ciertas propiedades y principios).
- b. E es una relación multívoca, reflexiva y transitiva en la familia de conjuntos C .
- c. La relación E refleja la emergencia de ciertas propiedades, esto es, la aparición de sistemas que son cualitativamente distintos de los que los forman."

A partir de esta definición, llamaremos nivel a cada uno de los conjuntos de sistemas de N . Como podemos ver, ésta es una formalización que se corresponde muy bien con la idea intuitiva de nivel jerárquico. La condición a. nos asegura que el nivel que estamos definiendo tenga un contenido semántico, entendido como una semejanza entre sus componentes. Es importante el concepto de emergencia, que es el verdadero núcleo de la definición. Para que aparezca un nivel superior no es suficiente que haya agregación, sino que es preciso que además aparezcan ciertas propiedades nuevas, no explicables (al menos de una forma sencilla y directa) en función de las propiedades de los sistemas componentes.

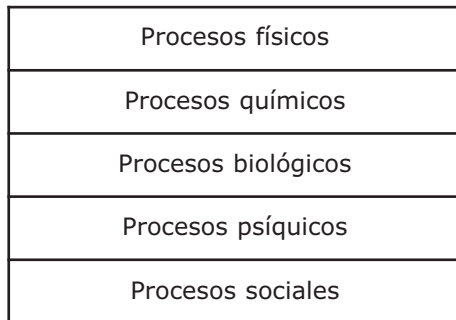


Fig. 4. Ejemplo de estructura de niveles sin relación de dominio

El concepto que hemos definido como estructura de niveles es independiente del de jerarquía. En concreto, no encontramos en la definición de estructura de niveles las ideas de dominación ni de orden (la relación de emergencia E no es asimétrica). Así, aunque pueda haber estructuras de niveles que sean jerárquicas, también puede haberlas que no lo sean.

Un ejemplo de estructura de niveles sin relación de dominación podría ser la de los procesos en la naturaleza (ver figura 4).

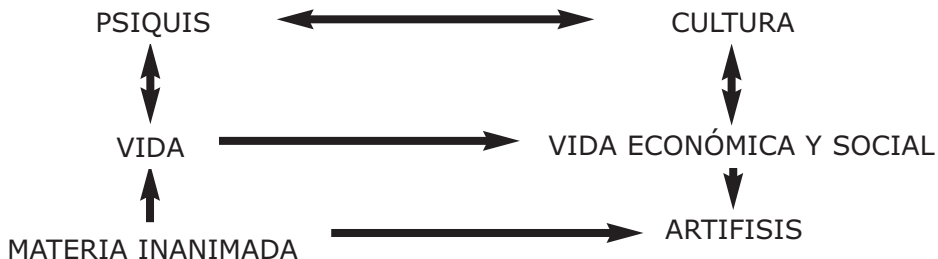


Fig. 5. Ejemplo de estructura de niveles no ordenada (fusión de dos estructuras de niveles: la de la naturaleza y la de la sociedad), tomado de [Bunge, 1969], pág. 38.

Y como estructura de niveles que ni siquiera está ordenada, podemos considerar la que propone Bunge para la naturaleza y la sociedad [Bunge, 1960] (ver figura 5). Sin pararnos en su significado, quedémonos sólo con su estructura, y veamos que se corresponde a la definición de estructura de niveles. C sería el conjunto formado por los seis niveles (psiquis, cultura, vida, vida económica y social, materia inanimada y artífisis), y E la relación dada por las flechas. A partir de aquí es fácil comprobar que se cumplen para este conjunto C y esta relación E las tres propiedades de la definición.

Invitamos al lector a reflexionar sobre la descomposición estructurada, representada incluso gráficamente, que del tema **Complejidad y Tecnologías de la Información** estamos haciendo en estas **Notas**:

- **Complejidad y Tecnologías de la Información:** G; M; T.I..
- **G:** Enfoques y Teorías; Sistemas; Tratamiento.
- **Enfoques y Teorías:** Variedad; Conceptos; Marcos.
- **Etc.**

7. Principios de la realidad jerárquica

A modo de conclusiones, exponemos a continuación los principios que propone Wells sobre la forma en que la jerarquía se manifiesta en el mundo real [Wells, 1969].

- a. La realidad está estructurada de forma jerárquica. El primer nivel de esta estructura estará compuesto por las partículas elementales. Luego vienen los de átomos y moléculas. Con esto se acaban los estratos inorgánicos. Pero continuamos con los que forman las células y los seres vivos multicelulares. Y aún podemos seguir con el estrato que componen los sistemas sociales.
- b. Los elementos de un nivel dado no pueden estar formados por un número cualquiera de entidades del nivel inferior, sino que existe una limitación en ese número. Así, las moléculas por lo general están compuestas por un número limitado de átomos (lo que las mantiene en un tamaño microscópico). Los propios átomos parecen presentar un límite en el número de electrones que contienen (que está en torno a 100). Esto también es válido para las estructuras sociales: los militares, por ejemplo, aplican los valores que consideran óptimos en el tamaño de los pelotones, de las compañías, etc. Esta idea podría expresarse en función de la apertura definida por Simon: "La apertura que pueden alcanzar los sistemas reales es limitada".
- c. En cualquier sistema jerárquico existen relaciones de complementariedad. Estas son notables en los enlaces químicos, que se basan en la unión de átomos complementarios en cuanto a su "apetencia" de electrones. Las organizaciones sociales pueden alcanzar objetivos imposibles de lograr aisladamente por las personas que las componen gracias a la diferenciación y complementariedad de las funciones que desempeñan estas personas.

- d. En un sistema integrado, todas las partes son necesarias. No podemos prescindir de ningún subsistema sin que el funcionamiento total se resienta, o incluso desaparezca. Aquí estriba una de las diferencias entre simple agregación de elementos y formación de un sistema. En el primer caso no hay orden, ni propiedades emergentes, mientras que en el segundo aparecen propiedades nuevas, que de alguna manera dan entidad diferenciable al todo.
- e. Las entidades libres son las que dan dinamismo al universo. Llamamos aquí entidades libres a las que tienen mayor complejidad dentro de su nivel, y no están incluidas en ninguna entidad de nivel superior. Tenemos así átomos libres (cuando no forman moléculas), moléculas libres, células libres... Al no estar "encajadas" en un lugar fijo dentro de un sistema, están listas para una integración, con lo que se posibilita la aparición de nuevos tipos de sistemas.

8. Importancia del concepto de jerarquía

Como ya hemos ido viendo por los ejemplos expuestos, las estructuras jerárquicas son muy abundantes en el mundo real. ¿Qué es lo que hace que este tipo de organización se halle tan extendida? En este apartado vamos a intentar explicarlo en función de su utilidad en tres grandes áreas:

1. Construcción de complejidad.
2. Comprensión de complejidad.
3. Procesamiento de información.

8.1 Construcción de complejidad

Para construir un sistema de una forma jerárquica podemos seguir el siguiente proceso:

- a. Partimos de elementos simples.
- b. Ensamblamos estos elementos, para formar un sistema de mayor complejidad.
- c. Considerando los sistemas obtenidos en el apartado b., los consideramos ahora como elementos básicos y pasamos de nuevo al punto a., hasta que consigamos un sistema de la complejidad deseada.

Para comprobar las ventajas que supone el proceso descrito bastará con recurrir a un ejemplo propuesto por Simon [Simon, 1962]. Tempus y Hora son dos relojeros. Pero mientras que Tempus ensambla los relojes montando los elementos de uno en uno, Hora lo hace por módulos, de forma jerárquica. Cada vez que uno de los

relojeros recibe a un cliente, ha de atenderle, con lo que pierde todo el trabajo que estaba realizando en ese momento. La diferencia está en que mientras Hora como mucho puede perder, por cada llamada de un cliente, el tiempo empleado en montar un módulo, Tempus debe rehacer el reloj completo.

De este ejemplo se deduce fácilmente la razón por la cual la estructura jerárquica permite trabajar del modo propuesto, mientras otras formas de organización, no. Y es que en el caso jerárquico hay formas intermedias (los módulos de Hora) que son estables y "cerradas". Y estas formas permiten continuar el proceso de ensamblado ocultando gran parte de su complejidad.

8.2 Comprensión de la complejidad

Ya hemos visto que la estructura jerárquica nos permite abordar el problema de representar los sistemas complejos con detalle y sencillez a la vez. Con detalle, porque cada nivel (o cada "módulo" parte de un nivel, puede ser descrito con tanto detalle como queramos. Y de una forma relativamente sencilla de comprender porque nos podemos centrar por un lado en comprender la estructura jerárquica del sistema, y por otro la de cada uno de los "módulos" que lo componen. Y la complejidad de estas tareas es mucho menor que la que implica la comprensión del sistema de una forma directa.

Un sistema es descomponible cuando sus partes pueden aislarse unas de otras. Y cuasidescomponible cuando los subsistemas que lo componen pueden considerarse casi aislados, por ser las relaciones entre ello mucho más débiles que las que se dan en su interior. Simon menciona estas propiedades de descomponibilidad o cuasidescomponibilidad de los sistemas jerárquicos como las que facilitan la comprensión [Simon, 1962]. El concepto de cuasidescomponibilidad es muy útil como principio de diseño de sistemas, como se ha demostrado en [Sáez-Vacas et al., 1982].

8.3 Procesamiento de información

En general, podemos decir que los excesos en la cantidad de información que un sistema debe manipular son nocivos. Desde luego, serán costosos en utilización de recursos, y requerirán un esfuerzo extra. Pues bien, la organización jerárquica también es eficiente desde este punto de vista. Es muy económica en el flujo de información necesario para la coordinación y funcionamiento [Milsum, 1978]. De esta forma reduce el coste y la posibilidad de que se produzcan sobrecargas.

Otro aspecto destacable es que la estructura jerárquica permite que gran parte de la información se procese de forma distribuida por todo el sistema. Ciertas partes pueden especializarse en algunas tareas, liberando así al resto de realizarlas. Además, al trabajar de esta forma, pueden actuar como filtros de información, permitiendo que sólo pase al sistema la estrictamente necesaria.

9. Resumen

Hay diferentes formas de entender la idea de jerarquía, pero todas incluyen el concepto de **clasificación en estratos**, según algún **rango** (característica de los objetos clasificados).

Los sistemas jerárquicos pueden definirse recursivamente como un conjunto de subsistemas **jerárquicos interrelacionados**.

En las jerarquías estructurales, tal como las entiende Simon, es importante la idea de **emergencia** de nuevas propiedades y comportamientos al pasar de un nivel a otro superior.

El concepto de jerarquía se aplica al menos en tres formas relacionadas: en la descripción de la realidad (**jerarquía de estratos**), en la toma de decisiones (**jerarquía multicapa de toma de decisiones**) y en la organización (**jerarquía de organización**).

Dada la formulación matemática del concepto de jerarquía, puede definirse el concepto de **estructura de niveles**, más amplio, y que engloba al anterior.

La importancia de las estructuras jerárquicas viene dada, entre otras ventajas que aportan, por su facilidad para **construir complejidad**, para **comprenderla**, y para **procesar información**.

El concepto de jerarquía, desde el punto de vista del análisis de la variedad, podría entenderse en cierto modo como un **filtro de variedad** (ver "Adaptación = Amplificación + Reducción", en el capítulo "La variedad"). Por una parte está una parte del mundo real, de una gran variedad. Por otra, un observador, con sus limitadas capacidades. Pues bien, el usar una aproximación jerárquica le permite tratar con aspectos de variedad manejable. Por una parte, mientras está estudiando un determinado nivel, sólo se enfrenta con su variedad (en general mucho menor que la asociada al objeto entero). Y por otra, cuando busca una visión de conjunto, sólo tiene que contemplar el nivel más alto, lo que de nuevo le permite tratar con algo de relativamente poca variedad.

10. Bibliografía

Dividida en dos apartados. En el apartado de Notas Bibliográficas se comentan aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

Un libro muy adecuado para una toma de contacto con el concepto de jerarquía es "**Las estructuras jerárquicas**", constituido por las actas del simposio celebrado en noviembre de 1968 en los Douglas Advanced Research Laboratories. Comienza con un repaso general y teórico de las jerarquías, para detenerse luego en campos más concretos y prácticos (las jerarquías orgánicas, las inorgánicas, y las de las artificiales). Para un lector novato en estos temas, será de especial interés la primera parte (repaso general), y especialmente los artículos de Whyte [Whyte, 1969] (donde se hace una revisión histórica de la evolución del estudio de las jerarquías), Bunge [Bunge, 1969] (donde se desarrolla una formalización matemática de conceptos relacionados con la jerarquía), y Mesarovic y Macko [Mesarovic y Macko, 1969] (de donde hemos tomado la distinción entre jerarquía de descripción, de toma de decisiones y de organización).

Referencias Bibliográficas

Bunge, M. (1960): "On the connections among levels", **Proceedings of the XIIth International Congress of Philosophy**, vol. 6, Sansoni, Florencia.

Bunge, M. (1969): "La metafísica, epistemología y metodología de los niveles", publicado en Whyte, L.: "**Hierarchical structures**", American Elsevier [en español (1973): "Las estructuras jerárquicas", Alianza, Madrid].

Grene, M. (1969): "La jerarquía: una palabra, pero ¿cuántos conceptos?", publicado en Whyte, L.: "**Hierarchical structures**", American Elsevier [en español (1973): "Las estructuras jerárquicas", Alianza, Madrid].

Mesarovic, M.D. y Macko, D. (1969): "Fundamentos de una teoría científica de los sistemas jerárquicos", publicado en Whyte, L.: "**Hierarchical structures**", American Elsevier [en español (1973): "Las estructuras jerárquicas", Alianza, Madrid].

Milsum, J.H. (1978): "**La base jerárquica para los sistemas generales vivientes**", publicado en Klir, G.J.: "Tendencias en la teoría general de sistemas", Alianza Editorial, Madrid.

Sáez-Vacas, F. y Lampaya D. (1982): "Concepción multinivélica y cuasidescomponible de sistemas complejos. Aplicación a la informática". **Actas V Congreso de Informática y Automática**, pp. 281-286.

Shapley, H. (1958): "**Of stars and men**", Beacon Press, Boston [en español (1963): "De estrellas y hombres", F.C.E., México.]

Simon, H. (1962): "**The architecture of complexity**", Proceedings American Philosophical Society, núm.106, pp.467-482 [en español (1979): "La arquitectura de la complejidad" publicado en Simon, H.: "Las ciencias de lo artificial", pp.128-169, A.T.E., Barcelona].

Wells, B. (1969): "Los niveles y las entidades integradas", publicado en Whyte, L.: "**Hierarchical structures**", American Elsevier [en español (1973): "Las estructuras jerárquicas", Alianza, Madrid].

Whyte, L.L. (1969): "Las jerarquías estructurales, o una retadora clase de problemas físicos y biológicos", publicado en Whyte, L.: "**Hierarchical structures**", American Elsevier [en español (1973): "Las estructuras jerárquicas", Alianza, Madrid].

Bloque II: El modelo H x I x O=IO

Capítulo 9: Textos sobre complejidad

En las páginas anteriores se ha desarrollado un conjunto de conceptos, teorías, métodos e ideas sobre la complejidad a un nivel desprovisto de formalizaciones elevadas. Por el contrario, se ha procurado más bien acentuar su tratamiento intuitivo.

Ahora entraremos en el planteamiento de un modelo muy general que recoja las esencias de una aproximación global al problema de la comprensión de "situaciones" complejas. Como preámbulo, en este capítulo seleccionaremos un conjunto de textos sobre complejidad y nociones conexas, que plantean, entre muy diversas cuestiones interesantes, una perspectiva de la vastedad de aspectos de la complejidad.

Estos textos, aparte de servir como pie al capítulo siguiente y principal, deberían suscitar en el lector una reflexión sobre diferentes cuestiones presentadas en anteriores capítulos.



1. Introducción

Este capítulo se compone de un ramillete de textos extraídos de la bibliografía sobre ciencia, filosofía, tecnologías de la información e informática, que por sí solos muestran la importancia y la riqueza (complejidad) de este asunto de la complejidad. Prácticamente, cada cita textual expresa una tesis, o casi.

2. El conocimiento, oculto detrás de la complejidad

Famosos pensadores, desde Pascal hasta Whitehead, pasando por Valery, Bachelard y otros muchos, nos dicen que el conocimiento sólo es posible dentro de la complejidad, y que para conseguirlo el método cartesiano ya no es suficiente, como no lo es tampoco un método exclusivamente globalista. Tal vez sea Edgar Morin el autor que mejor haya expresado modernamente el pensamiento de la complejidad a través de su enciclopédica y todavía inacabada síntesis en varios volúmenes, genéricamente titulada La Méthode.

Lo cierto es que la preocupación por la complejidad empieza a salir fuera de estos círculos reflexivos minoritarios. Pueden citarse varios ejemplos.

A mediados de los ochenta, el Club de Roma situó la complejidad del mundo como la principal causa para construir un tipo de enseñanza que promueva el aprendizaje innovador.

Otra manifestación indicativa es que la Association for Computing Machinery, una de las dos asociaciones profesionales de Informática más importantes, que organiza al cabo del año varias decenas de congresos muy especializados, inicia a finales de 1990 una serie de conferencias anuales sobre temas críticos en la informática: el primero ha versado sobre "Managing Complexity".

Y ya, para acabar, es pertinente saludar la aparición de libros, tal como uno recientemente publicado y escrito por un científico prestigioso, H.R.Pagels, que extienden a un público más amplio algo de ese pensamiento representado por Morin y por varios de los autores recogidos en este capítulo. Para nosotros es significativo el subtítulo del referido libro de Pagels, "The Computer and the Rise of the Sciences of Complexity", porque conjuga los dos pilares que nos interesan aquí: la tecnología y la complejidad.

No existe lo simple, sólo lo simplificado (**Bachelard**).

Lo simple no es más que un momento arbitrario de abstracción, un medio de manipulación arrancado a las complejidades (**Morin**).

El nuevo reino del conocimiento es, aquí, el de la complejidad (**Morin**).

Todo lo que es simple, es falso; todo lo que es complicado, es inutilizable (**Valery**).

El estudio de la complejidad organizada es el problema clave de todas las ramas de la ciencia moderna (**Gravitz**).

Toda realidad es unidad compleja (**Whitehead**).

Siendo todas las cosas causadas y causantes, ayudadas y ayudantes, mediatas e inmediatas, y relacionándose todas por un lazo natural e insensible que liga las más alejadas y las más diferentes, tengo por imposible conocer las partes sin conocer el todo, tanto como conocer el todo sin conocer particularmente las partes (**Pascal**).

Reduccionista u "holística" (globalista), la explicación, en uno y otro caso, busca simplificar el problema de la unidad compleja. Una reduce la explicación del todo a las propiedades de las partes concebidas aisladamente. La otra reduce las propiedades de las partes a las propiedades del todo, igualmente concebido como algo aislado (**Morin**).

La gran frontera inexplorada es la complejidad (...) Estoy convencido de que las naciones y personas que dominen las nuevas ciencias de la complejidad se convertirán en las superpotencias económicas, culturales y políticas del próximo siglo (**Pagels**).

Nuestros instrumentos nos han mostrado el mundo visible e invisible del mundo de la materia -átomos, moléculas, protones, y células; Sabemos lo que hay ahí. Lo que no sabemos es cómo está organizado, en detalle: un problema de complejidad. Si los tres primeros siglos de la ciencia moderna han extendido el sensorio humano, aprendido las propiedades de la materia y la vida, los tres próximos verán la emergencia de la complejidad (**Pagels**).

El informe al Club de Roma "Aprender, horizonte sin límites" define el aprendizaje innovador como el instrumento para salvar la distancia que media entre la creciente complejidad del mundo y nuestra capacidad para hacerle frente (**Club de Roma**).

2.1 Subjetividad en la complejidad

La naturaleza poliédrica -aunque por su carácter geométrico de estructura ordenada ni siquiera la voz 'poliédrica' termine de ser la expresión adecuada- del concepto de complejidad hace realmente poco factible una clasificación adecuada de los textos seleccionados, que, por lo demás, expresan con frecuencia lógicas diferentes. No obstante, necesidades prácticas nos han llevado a agruparlos en apartados y titularlos de una manera aproximada, eligiendo en cada caso el criterio que nos ha parecido más significativo.

Este apartado recoge varias citas en las que queda muy patente el condicionamiento sobre la complejidad por parte del observador, del sujeto que define o utiliza un objeto o su conocimiento de él. Sus percepciones, sus métodos, sus instrumentos, sus construcciones mentales, el lenguaje empleado, su cultura, su visión del mundo (Weltanschauung), su voluntad, su interés o el punto de vista elegido se confabulan para producir, no la realidad, sino un fragmento de ella, una imagen, más o menos conveniente o abarcable, de la realidad. Esto es, la complejidad no es una propiedad absolutamente objetivable.

El método no puede separarse de su objeto (**Heisenberg**).

Cuenta Needham la ironía con la cual los letrados chinos acogieron, en el siglo XVIII, el anuncio por los jesuitas de los triunfos de la ciencia moderna; la idea de que la naturaleza podía estar sometida a leyes simples y conocibles constituía para los mandarines un ejemplo de ingenuidad antropomórfica (**Prigogine, Stengers**).

Hay que saber que la información no existe. No es más que un desvío teórico para comprender lo real (**Attali**).

La percepción de los sentidos está regida por mecanismos que hacen nuestro conocimiento del mundo exterior altamente inferencial. No recibimos impresiones que sean elementales. Nuestras impresiones sensoriales están ellas mismas construidas por el sistema nervioso, de tal modo que, automáticamente, llevan consigo una interpretación de lo que ven, oyen o sienten (**Bronowsky**).

El ser vivo es en parte una fábrica química, en parte una máquina calculadora, en parte un alma pensante... Estas representaciones se complementan, pero ninguna agota el asunto (**Brillouin**).

Los hechos son mudos: sin una preconcepción no se puede saber cuáles son datos relevantes al propósito investigador y cuáles no lo son; cada instrumento de observación lleva implícitas unas hipótesis sobre la naturaleza de la realidad investigada (...) De ello se desprende que la objetividad científica absoluta es teóricamente imposible; toda teoría o programa de investigación lleva en su raíz una decisión subjetiva, que favorece ciertos aspectos e ignora otros (**Racionero**).

Cuan compleja o simple sea una estructura depende críticamente de la forma en que la describamos (**Simon**).

La definición de un objeto se hace por triangulación: pondera una definición funcional (lo que hace el objeto), una definición ontológica (lo que el objeto es) y una definición genética (lo que el objeto deviene) (**Le Moigne**).

¿Quién soy yo? Puedo concebirme como un sistema físico de miles de billones de átomos; un sistema biológico de treinta mil millones de células; un sistema orgánico de centenas de órganos; un elemento de mi sistema familiar, o urbano, o profesional, o social, o nacional, o étnico...(**Morin**).

¿Qué es más simple, un círculo o un polígono? Depende de cómo se mire. Desde un punto de vista orientado a los números, un polígono es simple y un círculo es complejo: un polígono se obtiene por el fácil método de enlazar juntos un cierto número de segmentos lineales iguales, mientras que un círculo emerge sólo como un límite ideal -un polígono con un número infinito de lados infinitesimales. Desde un punto de vista orientado al espacio, podemos usar

ecuaciones algebraicas para hablar de curvas espaciales y nos encontramos con que el círculo tiene una fórmula muy simple, mientras que un polígono tiene una clase de fórmula mucho más compleja, con muchos términos, uno para definir cada uno de los lados del polígono (**Rucker**).

En ciertos momentos, bajo ciertos ángulos, la parte puede ser más rica que la totalidad (**Morin**).

Hay siempre en la extracción, delimitación, definición de un sistema algo incierto o arbitrario: hay siempre decisión y opción, lo que introduce en el concepto de sistema la categoría del sujeto. El sujeto interviene en la definición del sistema en y por sus intereses, sus selecciones y metas, es decir, que él aporta al concepto de sistema, a través de su sobredeterminación subjetiva, la sobredeterminación cultural, social y antropológica (**Morin**).

Al cambiar lo que conoce del mundo, el hombre cambia el mundo que él conoce; al cambiar el mundo en el que vive, el hombre se cambia a sí mismo (**Dobshzhansky**).

La tragedia de la complejidad se sitúa a dos niveles, el del objeto del conocimiento y el de la obra de conocimiento. Al nivel del objeto, estamos situados incesantemente ante la alternativa, por un lado, del cierre del objeto del conocimiento, que mutila sus solidaridades con los otros objetos así como con su propio entorno (y que excluye de golpe los problemas globales y fundamentales) y, por otro lado, de la disolución de las fronteras y contornos que anega todo objeto y nos condena a la superficialidad. Al nivel de la obra, el pensamiento complejo reconoce a la vez la imposibilidad y la necesidad de una totalización, de una unificación, de una síntesis. Por tanto, debe afrontar trágicamente la totalización, la unificación, la síntesis, luchando al tiempo contra la pretensión a esa totalidad, a esa unidad, a esa síntesis, en la conciencia plena e irremediable del inacabamiento de todo conocimiento, de todo pensamiento y de toda obra (**Morin**).

Preguntas como ¿qué es más complejo, un círculo o un polígono, una rosa o tal teorema matemático? carecen de sentido. En un objeto cualquiera podemos ver mil objetos distintos, como de manera muy didáctica expresa el ejemplo de las cinco formas matemáticas de modelar la mano humana, tomado de un delicioso libro de Rucker. En su extremo, como muy bien dice Morin, la complejidad nos plantea el problema trágico de los límites del conocimiento y de la acción.

Tomemos un objeto específico y mostremos cómo puede ser pensado, en tanto que patrón matemático, en cinco formas diferentes. Usemos para ello la mano derecha.

1. **La mano, como número.** En el nivel más superficial, una mano es un ejemplo del número 5. Fijándose en los detalles, uno advierte que la mano tiene un cierto número de pelos y de surcos. Los dedos tienen longitudes numéricas específicas que pueden medirse en milímetros. Puede calcularse el área de cada una de las uñas, así como su masa. Mediciones internas de la mano pueden producir muchos más números: temperaturas, tasas de flujo sanguíneo, conductividad eléctrica, salinidad, etc. La mano codifica un buen montón de números.
2. **La mano, como espacio.** La mano es un objeto en un espacio tridimensional. No tiene agujeros y está conectada a nuestro cuerpo. La superficie curvada bidimensional de la piel es convexa en algunas regiones y cóncava en otras. Los vasos sanguíneos forman una ramificación unidimensional. La huella de su pulgar es aproximadamente elipsoidea y sus dedos parecen conos truncados. Las uñas son paraboloides aplanados y sus células epiteliales son cilíndricas. La mano es un muestrario de formas espaciales.
3. **La mano, como lógica.** Los músculos, huesos y tendones de la mano constituyen una clase de máquina, y las máquinas son especiales suertes de patrones lógicos. Si se tira de este tendón por aquí, se mueve aquel hueso de ahí. Aparte la cuestión mecánica, la mano posee varias pautas de comportamiento que se conjuntan en una forma lógica. Si nuestra mano toca el fuego, se retira bruscamente. Si toca un conejito, lo acaricia. Si aprieta, sus nudillos se vuelven blancos. Si andan en el lodo, sus dedos ennegrecen. Nuestro conocimiento lógico de la mano podría llenar un pesado manual.
4. **La mano, como infinitud.** Hablando en abstracto, la mano contiene infinitos puntos espaciales. A título práctico, escalas progresivamente más pequeñas revelan más y más estructura. Mirada de cerca, la superficie de la piel es un patrón complejo y sin fin del tipo conocido como "fractal". Lo que conocemos de nuestra mano se relaciona con lo que conocemos acerca de una red de otros conceptos -es difícil despegar nuestra mano del mar infinito de todo el conocimiento. Otra clase de infinitud proviene del hecho de que nuestra mano es parte de nosotros, y la esencia viva de una persona está fuertemente relacionada con las infinitudes paradójicas de la teoría de conjuntos (la versión de los matemáticos de la teología).
5. **La mano, como información.** La mano está diseñada de acuerdo con ciertas instrucciones codificadas en nuestro ADN. La longitud de estas instrucciones da una medida de la cantidad de información en nuestra mano. En el curso de la vida nuestra mano se ha visto sometida a diversas influencias que han dejado cicatrices, pecas, etc. Podríamos desear incluir estas influencias en nuestra medida de la información sobre la mano. Una manera de hacerlo consistiría en asociar el contenido de información sobre la mano con el número de preguntas a las que habría que contestar para poder construir una réplica de esa mano. Otra manera sería estimar la longitud del programa de ordenador más corto que fuera capaz de responder a toda pregunta relacionada con esa mano (**Rucker**).

2.2 Aspectos de la complejidad

Las citas que hemos seleccionado en este apartado son una breve muestra de los múltiples aspectos que surgen en el estudio o en la gestión de la complejidad.

Destacan cuestiones como su relación con el indeterminismo, el desorden, la imprecisión y hasta el azar. El humorista Perich, a la pregunta de ¿qué es el azar? respondía: "el motivo de que las cosas nos salgan mal, no como cuando nos salen bien que se debe a nuestro talento". Más poético, pero no por eso más penetrante, resulta Borges, cuando dice: el azar, tal es el nombre que nuestra inevitable ignorancia da al tejido infinito e incalculable de causas y efectos.

Bromas aparte, es evidente que a esa concepción de la complejidad se añade otra, bien distinta y tan importante como la primera, que la considera como una máscara (un código desconocido) de la simplicidad o al menos del orden.

A veces, ni siquiera hay desorden sino que, al revés, el objeto se nos puede aparecer como una muy ordenada pero intrincada maraña formada por patrones muy elementales, como sucede con los dibujos islámicos, tanto florales (tauriq) como geométricos (tastir), cuya significación y reglas de estructuración pertenecían al orden secreto de lo religioso. Los modernos mosaicos de Penrose, los dibujos y grabados de Escher, las formas de Ostwald y los fractales de Mandelbrot son diversas demostraciones en los dominios de la matemática, la física y la plástica de estructuras complejas generadas a partir de formas extremadamente simples. Dichas demostraciones sugieren un modelo de complejidad, pero de ninguna manera agotan su contenido.

Podemos conocer más de lo que nos es posible decir, y, por eso mismo, una rigurosa lucidez que pretenda precisar todos los detalles puede impedir la comprensión de temas complejos: al analizar los componentes vagamente conocidos, destruimos el conjunto que forman y perdemos el conocimiento del conjunto (**Racionero**).

Nos encontramos en un mundo azaroso, un mundo en el cual la reversibilidad y el determinismo son solamente aplicables a situaciones límites y casos simples, siendo al contrario la regla la irreversibilidad y la indeterminación. (...) Queremos mostrar que la irreversibilidad surge cuando los conceptos básicos de las Físicas clásica y cuántica, como trayectorias y funciones de onda, corresponden a idealizaciones excesivas. No es que las leyes fundamentales de las Físicas clásica y cuántica sean falsas, sino que representan en algunos casos una simplificación desmedida, inaccesible a cualquier experimento concebible (**Prigogine, Stengers**).

La complejidad es un desorden aparente, en el que hay razones para suponer un orden oculto. La complejidad es un orden cuyo código no se conoce (**Atlan**).

Ninguna cosa o ser organizado puede escapar a la degradación, la desorganización, la dispersión (**Morin**).

No hay organización sin anti-organización. La anti-organización es a la vez necesaria y antagonista a la organización. Para la organización fija, la anti-organización es virtual, latente. Para la organización activa, la anti-organización deviene activa (**Morin**).

El crecimiento de complejidad y el crecimiento de desórdenes están ligados (**Morin**).

A todo crecimiento de complejidad en la organización le corresponden nuevas potencialidades de desorganización (**Morin**).

Toda concepción ideal de una organización que no fuera más que orden, funcionalidad, armonía, coherencia es un sueño demente de ideólogo o de tecnócrata. La racionalidad que eliminase el desorden, la incertidumbre, el error no es otra cosa que la irracionalidad que eliminaría la vida (**Morin**).

Juntado lo que está completo y lo que no, lo que concuerda y lo que no, lo que está en armonía y lo que está en desacuerdo (**Heráclito**).

Es sorprendente, pero cierto, que la complejidad del mundo es una consecuencia de la combinación de unas seis docenas de átomos diferentes (**Pagels**).

La tarea central de la ciencia natural es mostrar que la complejidad, correctamente considerada, es sólo una máscara de la simplicidad; encontrar patrones escondidos en el aparente caos (**Simon**).

Sucede que para algunos sistemas complejos hay una simplicidad subyacente -sólo unas pocas variables son realmente importantes. Puede demostrarse que la interacción de unos pocos componentes de acuerdo con un conjunto de reglas produce fenómenos complejos. Quizá todos esos miles de variables son sólo superficiales, y en su base las cosas son muy simples. Pero hasta que esa simplicidad, si es que existe, sea puesta de manifiesto, tendremos que manejar directamente la complejidad. Afortunadamente, esto ya es posible, gracias al ordenador (**Pagels**).

El pensamiento complejo no enfoca la "totalidad", en el sentido en que este término sustituye una simplificación atomizante por una simplificación globalizante, en la que a la reducción a las partes sucede la reducción al todo. (...) El pensamiento complejo afronta, no lo elemental -donde todo se funde en la unidad simple y el pensamiento claro- sino lo radical, donde aparecen incertidumbre y antinomias (**Morin**).

(...) complejidad por superposición de simplicidades, una complejidad que sólo percibo en su globalidad, pero que estoy muy lejos de describir satisfactoriamente. Existen complejidades, es verdad, a cuya descripción debo renunciar (**Wagensberg**).

El problema de la definición precisa de la noción de complejidad como concepto científico fundamental (análogo a los de energía, entropía, etc) está todavía planteado (**Atlan**).

Por consiguiente, no existe ninguna droga que no tenga efectos secundarios. Y cuanto más prestigiosa sea la receta (antibióticos, cortisona, tranquilizantes, compuestos contra la hipertensión, relajadores musculares), mayor resulta el problema de los efectos secundarios adversos. Los medicamentos pueden alterar o modificar los equilibrios del torrente sanguíneo. Pueden hacer que la sangre coagule más o menos rápidamente. Pueden reducir el nivel de oxígeno de la sangre. Pueden estimular el sistema endocrino, aumentar el flujo de ácido hidrocórico hacia el estómago, o acelerar o disminuir el paso de sangre a través del corazón, desequilibrar la producción de sangre del cuerpo reprimiendo la médula ósea, reducir o aumentar la presión sanguínea o afectar al intercambio entre sodio y potasio que juega un papel vital en el equilibrio químico del cuerpo (**Cousins**).

Con motivo de la concesión del Premio Nobel de Física de 1990 a unos investigadores que colaboraron destacadamente en la elaboración del modelo de los quarks, la Academia Sueca nos recordaba que éstas son las partículas más elementales conocidas. Toda la materia del universo, incluido el cuerpo humano, está formada en más de un 99% por quarks y gluones (partículas que unen los quarks) y el resto son electrones. Estas partículas no son formas muy simples, pero aunque lo fueran, tal circunstancia sólo nos adelanta un ápice para conocer la complejidad del hombre y del universo en la inmensa mayoría de las situaciones que nos interesan.

En líneas generales, es necesario aceptar que la complejidad tiene componentes de orden y de desorden, que imbrica la organización y la anti-organización, que implica armonía y desacuerdo, que genera efectos positivos y negativos: un ejemplo tomado de la fisiología lo pone bien de manifiesto. Cuando sólo se aprecia una vertiente, es porque la otra está latente, salvo que estemos hablando de construcciones teóricas o artificiales, como los citados arabescos.

El observador siempre está obligado a elegir entre los muchos aspectos de la complejidad de su objeto.

3. La sobrespecialización, causa y efecto de la complejidad

La fragmentación del conocimiento en numerosas disciplinas es una consecuencia de los progresos de la ciencia y de la cultura, lo que ha dado origen a una frase célebre del poeta Eliot relativa al exceso de información y la pérdida del saber: "¿adónde se fue el saber que hemos perdido en el conocimiento, adónde el conocimiento que hemos perdido en la información?". El mismo sentido tiene esta otra, más escueta, de Daumel: "lo sé todo, pero no comprendo nada".

Es una evidencia que la multiplicación de especialidades tiene el efecto de facilitar la división del trabajo y la formación, pero produce una dificultad geoméricamente creciente en lo que atañe a afrontar la realidad compleja que nos circunda. Si la interdisciplinariedad es conveniente, aunque cada día más difícil de conseguir, ¡qué decir entonces de la multidisciplinariedad!.

Ante un exceso insostenible de especialización comienzan también a multiplicarse -aunque todavía en número relativamente modesto en el plano práctico- las voces que reclaman difundir una "ciencia de la complejidad". Desgraciadamente, la estructura oficial del conocimiento y del mundo laboral e industrial, arrastrados por una inercia de decenas y decenas de años se muestran muy cerrados a estos nuevos/viejos planteamientos.

El conocimiento aislado obtenido por un grupo de especialistas en un campo estrecho no tiene en sí mismo valor alguno de ninguna clase. Sólo tiene valor en el sistema teórico que lo une a todo el resto del conocimiento (**Schrödinger**).

En quince años el número de especialidades reconocidas por la Asociación Médica de los Estados Unidos ha aumentado a más del doble y en la actualidad (1975) comprende sesenta y siete campos. Dentro de cada campo se forma un feudo, en el que se reconocen enfermeras, técnicos, revistas, congresos y algunas veces grupos organizados de pacientes que pugnan por obtener más fondos públicos. El costo de coordinación del tratamiento del mismo paciente por varios especialistas crece exponencialmente con cada competencia que se agrega en el proceso, y lo mismo ocurre con el riesgo de que se cometan errores y la probabilidad de provocar lesiones a causa de la inesperada sinergia de diferentes terapéuticas (**Illich**).

La rarefacción de las comunicaciones entre ciencias naturales y ciencias humanas, la disciplinariedad cerrada (apenas corregida por la insuficiente interdisciplinariedad), el crecimiento exponencial de los saberes separados hacen que cada uno de nosotros, especialista o no, sea cada día más ignorante del saber existente (**Morin**).

La nueva ciencia de la complejidad, como el computador, atraviesa las fronteras disciplina-rias. Algunos problemas en Biología, Física, Ingeniería eléctrica, Economía y Antropología se pueden tratar por medio de métodos similares, no por causa de solapamiento alguno en

cuanto a la naturaleza de esos campos, sino porque las técnicas requeridas para resolverlos son similares. La futura organización de las ciencias, ejemplarizada por la forma en que los departamentos científicos son categorizados en las universidades, cambiará para reflejar esta nueva estructura interdisciplinaria de los problemas. Eso sucederá, o si no, emergerán nuevas instituciones que asumirán esta nueva arquitectura de las ciencias (...) De hecho, esto ya está ocurriendo en la medida en que se están estableciendo nuevos centros para el estudio de la complejidad dentro de algunas universidades o como institutos independientes (**Pagels**).

A medida que se vaya desarrollando nuestra comprensión de la complejidad, podrían encontrarse leyes de los sistemas complejos aplicables a una variedad de sistemas con independencia de la disciplina en la que estén localizados (**Pagels**).

De hecho, cabe esperar una resistencia considerable, porque la simplicidad es bastante más satisfactoria (más newtoniana) intelectualmente que la complejidad. Esta se toma habitualmente como una tara que hay que soportar, si no hay más remedio, pero que es preferible evitar cuando ello es posible. Y esta toma de postura no está ni mucho menos desprovista de razones, como demostró el famoso libro de Schumacher "Small is beautiful".

Como reflejo sarcástico de posiciones sociales muy generalizadas frente a la complejidad traemos a colación unos cuantos "aforismos" de un autor americano apellidado Gall que ha "sufrido" ciertas consecuencias de la complejidad de algunos sistemas creados por el hombre. ¿Fueron esos sistemas diseñados por conjuntos de gentes sobrespecializadas?.

Teoría avanzada de sistemas:

1. Todo es un sistema
2. Todo es parte de un sistema mayor.
3. El universo está infinitamente sistematizado, tanto en el sentido ascendente (sistemas mayores) como en el descendente (sistemas menores).
4. Todos los sistemas son infinitamente complejos (La ilusión de simplicidad se basa en focalizar la atención en una o en pocas variables).

Los programas de ordenador nunca funcionan a la primera. Los programas complejos nunca lo hacen.

Un sistema complejo puede fallar de un número infinito de formas.

Cuanto mayor es el sistema, más especializada y estrecha su interfaz con los individuos.

Los sistemas complejos tienden a producir respuestas complejas (no soluciones) a los problemas.

Nada complicado funciona. Los sistemas complicados raramente superan el 5% de eficiencia.

Los sistemas complejos tienden a oponerse a su propia específica función.

4. Necesidad de un nuevo discurso del método

Es preciso reconocer que los sarcasmos anteriores ilustran un estado intelectual deficitario, una carencia de conocimientos o de métodos, probablemente un desarme frente a la evolución extremadamente rápida del mundo, de sus instituciones, de sus sistemas sociales y de su tecnología. Ciertamente, reflejan la manifestación de excesos incontrolados y de falta de prudencia (o de sabiduría, como dirían Eliot y Fourastié, entre otros muchos).

Microscopio, telescopio: estas palabras evocan los grandes avances científicos hacia lo infinitamente pequeño y hacia lo infinitamente grande. (...) Hoy nos enfrentamos a otro infinito: lo infinitamente complejo. Pero esta vez no poseemos instrumento alguno. (...) Ciertamente es que el ordenador es un instrumento indispensable. Sin embargo, no es más que un catalizador, no todavía esa herramienta que tanto necesitamos. (...) El macroscopio no es una herramienta como las demás. Es un instrumento simbólico, hecho de un conjunto de métodos y de técnicas tomadas de muy diferentes disciplinas. (...) El macroscopio filtra los detalles, amplifica lo que une, destaca lo que aproxima. No sirve para ver más grande o más lejos, sino para observar lo que a la vez es demasiado grande, demasiado lento y demasiado complejo a nuestros ojos (como la sociedad humana, ese organismo gigantesco que nos es totalmente invisible). Antaño, para tratar de penetrar los misterios de la complejidad buscábase las unidades más simples que permitían explicarla: la molécula, el átomo, las partículas elementales (**Rosnay**).

LOS CUATRO PRECEPTOS DEL NUEVO DISCURSO DEL METODO (**Le Moigne**)

El precepto de pertinencia: Convenir en que todo objeto que consideremos se define por relación a las intenciones implícitas o explícitas del modelador. Jamás prohibir poner en duda esta definición si, habiéndose modificado nuestras intenciones, se modifica la percepción que teníamos del objeto.

El precepto de globalismo: Considerar siempre el objeto a conocer por nuestra inteligencia como una parte inmersa y activa en el seno de un todo mayor. Percibirlo primero globalmente, en su relación funcional con su entorno sin preocuparse exageradamente de establecer una imagen fiel de su estructura interna, cuya existencia y unicidad jamás se darán por conseguidas.

El precepto teleológico: Interpretar el objeto no en sí mismo, sino por su comportamiento, sin buscar explicar a priori su comportamiento por alguna ley implicada en una eventual estructura. Por el contrario, comprender este comportamiento y los recursos que moviliza por relación a los proyectos que, libremente, el modelador atribuye al objeto. Tener la identificación de estos hipotéticos proyectos por un acto racional de la inteligencia y convenir en que su demostración muy raramente será posible.

El precepto de agregatividad: Convenir en que toda representación es partidaria, no por olvido del modelador, sino deliberadamente. En consecuencia, buscar algunas fórmulas susceptibles de guiar la selección de agregados tenidos por pertinentes y excluir la objetividad ilusoria de un censo exhaustivo de los elementos a considerar.

La visión de la naturaleza ha sufrido un cambio radical hacia lo múltiple, lo evolutivo, lo complejo, una nueva situación que no tiene precedentes en la historia de la ciencia. (...) El interés de la ciencia se está trasladando desde lo simple a lo complejo. (...) ¿Cuáles son las hipótesis de la ciencia clásica de las cuales la ciencia actual se ha liberado? Fundamentalmente aquéllas que se centran alrededor de la convicción básica de que el mundo microscópico es simple y está gobernado por leyes matemáticas simples. Esto nos parece hoy día una idealización engañosa. Esta situación sería similar a la reducción de un edificio a un conglomerado de ladrillos; con los mismos ladrillos podríamos construir una factoría, un palacio o una catedral. Es al nivel del edificio en su conjunto donde vemos el efecto del tiempo y del estilo en el cual fue concebido el edificio (**Prigogine, Stengers**).

Ello no quita, sin embargo, para que la complejidad esté ahí, desafiándonos, exigiéndonos planteamientos y visiones diferentes, llámense macroscopios o nuevos discursos del método. Un problema es inventarlos o perfeccionarlos, y otro, no menor, y puede que incluso mucho mayor, cómo conjugarlos con el potente proceso de sobrespecialización hoy vigente.

4.1 Los sistemas, un instrumento

El concepto de "sistema" genera uno de los principales enfoques para el estudio de la complejidad. De hecho, nació para eso, bajo la pretensión de unificar el estudio de las ciencias. A nuestro parecer, es uno de los más eficaces instrumentos "cognitivos", y, como es fácil de comprobar, muchos de los autores citados en los apartados anteriores han hecho uso de él.

Existen muchas definiciones de "sistema", casi todas equivalentes a las dos categorías representadas en la tabla adjunta por las citas de Ashby y de Morin.

Los objetos pueden representar una infinidad de sistemas igualmente plausibles, que difieren los unos de los otros por sus propiedades (**Ashby**).

Sistema: unidad global organizada de interrelaciones entre elementos, acciones o individuos (**Morin**).

La complejificación de la organización social es necesaria en el marco de las sociedades relacionales. Complejificación significa apertura de un gran número de nuevos canales de comunicación para multiplicar las interdependencias entre subsistemas (**Attali**).

Principio de emergencia: El todo es más que la suma de las partes. Principio de constricción: El todo es menos que la suma de las partes (**Morin**).

Tanto en ciencia como en ingeniería, el estudio de los "sistemas" es una actividad crecientemente popular. Esto es más una respuesta a la necesidad opresiva de sintetizar y analizar la complejidad que a cualquier gran desarrollo de un cuerpo de conocimiento y técnica para afrontar la complejidad (**Simon**).

De manera más general, puede concebirse la evolución de sistemas organizados, o el fenómeno de auto-organización, como un proceso de aumento de complejidad a la vez estructural y funcional resultante de una sucesión de desorganizaciones recuperadas seguidas cada vez de un restablecimiento a un nivel de variedad mayor y de redundancia menor (**Atlan**).

Numerosas grandes empresas y administraciones son insuficientemente complejas: se presentan como un ensamblado demasiado complicado (múltiples códigos, múltiples normas, múltiples cadenas de control,...) de actividades y de procesos pobres y fragmentarios. Tales organizaciones sufren a la vez de un déficit global de variedad (en el sentido de la riqueza de percepciones, de los significados, de las comunicaciones,...) y de un insuficiente reparto de esta variedad entre las unidades de los diversos niveles: cada una de las unidades es simple (su función y su trabajo son a veces incluso simplistas) y su imbricación es complicada. Para el individuo alojado en una de estas unidades, ésta es su entorno próximo de trabajo y su entorno lejano, la fábrica, la empresa, el medio socio-económico, es indescifrable (**Mélèse**).

La noción de instrumentación es concebida de manera demasiado estrecha. Cuando pensamos en instrumentos pensamos en microscopios, telescopios, dispositivos electrónicos de medida: artefactos materiales. Sin embargo, las personas también crean "instrumentos cognitivos", especialmente técnicas matemáticas que pueden ser herramientas utilizadas por la mente humana para promover el descubrimiento (...) Otro ejemplo más actual de instrumento cognitivo es el software innovador (**Pagels**).

En cualquier caso, es recomendable una visión muy amplia de los sistemas, que incluya las constricciones y no sólo las emergencias, y que alcance a los sistemas evolutivos, como son los sistemas vivos y los sistemas sociales. A través de los fenómenos de organización, de auto-organización y de desorganización, la complejidad adquiere un sentido muy rico, que abarca también muchos de los aspectos positivos ligados a la evolución.

En efecto, nosotros hemos escrito hace tiempo lo siguiente: "El progreso tecnológico incrementa la complejidad organizativa y la Biología nos enseña que no hay evolución sin aumento de la complejidad interna. Es precisamente gracias a su enorme, inimitable y adecuada complejidad que la mano humana es entre todos los instrumentos mecánicos naturales y artificiales el más fino y multifuncional. Por lógica, los riesgos y problemas ocasionados por su pérdida o deterioro se miden en la misma escala de importancia".

Hoy es impropio estudiar la complejidad en relación con la tecnología si no se integra a ésta en un conocimiento de los sistemas humanos organizados e incluso de los sistemas biológicos.

4.2 Los modelos y las jerarquías

Se ha dicho que los sistemas no existen en la naturaleza, sólo son una aproximación con la que construimos modelos mentales. Estos modelos muchas veces adquieren diversas formas jerárquicas, porque éstas han demostrado su eficiencia para comprender o diseñar sistemas complejos.

El estudio de las jerarquías se ha convertido desde hace años en uno de los campos más fructíferos de la epistemología científica, y en el nivel práctico representa una indudable herramienta intelectual, una modalidad de abstracción inseparable del enfoque sistémico. En el campo de la tecnología su impronta se deja ver en cualquier sistema medianamente complejo, desde la estructura de un ordenador o de un sistema gestor de bases de datos hasta la arquitectura de una red informática de comunicaciones.

La visión del mundo que cada uno de nosotros se ha formado no es sino un modelo. Tomamos todas nuestras decisiones a partir de modelos. Todas las leyes son escritas fundándose sobre modelos (**Forrester**).

El concepto de sistema es, de hecho, inseparable del concepto de modelo, concebido como sistema representativo de un sistema concreto. Todo sistema real no es conocido en efecto más que a través de modelos representativos (representaciones mentales individuales o representaciones explicitadas formalmente) (**Walliser**).

Es evidente el inmenso alcance de la clasificación jerárquica: se trata del método de clasificación más poderoso que emplea el cerebro-y-entendimiento humano para poner orden en la experiencia, las observaciones, las entidades y la información; y, aunque no se ha asentado aún como tal de modo inconcuso en la neurofisiología y la psicología, este tipo de clasificación representa probablemente el modo primario de coordinación u organización: a) de los procesos corticales, b) de sus correlatos mentales, y c) de la expresión de éstos mediante simbolismos y lenguajes (**Whyte**).

El hecho de que muchos sistemas complejos tengan una estructura jerárquica cuasi-descomponible es un factor mayor que nos facilita la comprensión, la descripción y la "visión" de tales sistemas y de sus partes (**Simon**).

Entre posibles formas complejas, las jerarquías son unas que tienen el tiempo para evolucionar (**Simon**).

Casi por definición, los sistemas verdaderamente escapan a una descripción completa y detallada. Cuando hemos de describir algo, el dilema con que nos encontramos es el que existe entre la sencillez, que es uno de los requisitos previos de la comprensión y la necesidad de tener en cuenta los numerosos aspectos del comportamiento de un sistema complejo. Es posible intentar una vía de salida a tal dilema mediante una descripción jerárquica: se describe el sistema mediante una familia de modelos, cada uno de los cuales se ocupa de su comportamiento mirado desde un nivel de abstracción distinto. Entonces, para cada nivel existe un conjunto de rasgos, variables, leyes y principios pertinentes mediante las cuales describiremos el sistema en cuestión; ahora bien, para que semejante jerarquía sea efectiva es necesario que la descripción correspondiente a un nivel cualquiera se considere independiente de las relativas a los demás niveles, cosa que puede hacerse admitidas su clausura y su independencia frecuencial (**Bunge**).

Para el espíritu simplificador, la organización ideal procede de la estructura piramidal centralista/ jerárquica/ especializadora. En la cúspide, el centro de computación/ decisión/ mando. En escalones, las jerarquías de control, función, transmisión. En la base, los operadores especializados. (...) De hecho, cada vez que nuestra mirada toma amplitud o afina en el detalle, vemos que la combinación de centrismo/ policentrismo/ acentrismo, jerarquía/ heterarquía/ anarquía, especialización/ polivalencias/ no-especializaciones es el carácter finalmente fundamental de los fenómenos vivos (**Morin**).

Puede que no existan los sistemas en la naturaleza, pero en la tecnología, sí, porque son objetos artificiales, esto es, entidades construidas por el hombre, a imagen y semejanza de sus modelos mentales: sistemas y jerarquías, entre otros instrumentos cognitivos.

Referencias obligadas para estudiar las jerarquías son los libros de Simon, Whyte y Morin.

5. "Filosofía" e Ingeniería

A mediados de 1990, A. Penzias, vicepresidente de investigación en la empresa AT&T, decía lo siguiente: "La complejidad está creciendo a causa de que la tecnología nos capacita para hacer más cosas y nos fuerza a hacerlas mejor (...) Contrariamente a lo que podía esperarse, la complejidad de la tecnología nos está empujando a los ingenieros a asignarle un mayor valor a nuestras capacidades en relaciones humanas, porque tenemos que lidiar con más gente en el curso de nuestro trabajo".

La observación anterior es correcta, pero tal vez insuficiente. Muchos observadores piensan que el ingeniero debería preocuparse por variadas cuestiones filosóficas y epistemológicas, al menos una brizna más de lo que por lo regular lo hace ahora. Siendo así que ciertas aplicaciones de la informática, como la Inteligencia Artificial, modifican el concepto que de sí mismo pueda tener el ser humano ¿sería lógico evadirse completamente de esas y otras implicaciones para acantonarse en un reducto exclusivamente tecnológico?.

(...) los ingenieros no parecen ser conscientes de las cuestiones filosóficas que su trabajo pueda entrañar. Para entrar en conversación con mis amigos ingenieros a veces pregunto: ¿cuáles son los fundamentos de su disciplina?. La pregunta siempre es recibida con perplejidad. Incluso después de haber explicado qué es lo que me ocupa, esto es, un informe coherente de la naturaleza y significado de la rama de ingeniería en la que ellos trabajan, la pregunta carece de significado para ellos. Los muy pocos que formulan preguntas importantes acerca de sus profesiones técnicas por lo general son considerados por sus colegas maniáticos, peligrosos y radicales (**Winner**).

¿Quién tiene más necesidad de la epistemología? Son los ingenieros quienes tienen la necesidad más urgente de una teoría del conocimiento y la mejor probabilidad de creer en ella (**Papert y Voyat**).

Si el ingeniero falla en estudiar metainformática, o, peor aún, si permanece inconsciente del hecho de que hay límites a la aplicabilidad de la computación y de las tecnologías de la información, es probable que caiga en una clase de error similar a la de ciertos teólogos medievales que intentaban establecer cuestiones de física por medio de citas bíblicas (Paráfrasis de **Schumacher**).

Los técnicos suelen ofenderse cuando se les dice que están empapados de filosofía. (...) Por más que rehuya a los filósofos, el técnico no puede evitar el contagio filosófico, ya que maneja ideas que presuponen conceptos e hipótesis filosóficos: hace filosofía sin saberlo. Y, puesto que la hace, mejor sería que la hiciese bien. Para esto tendría que aprender algo de filosofía (**Bunge**).

En conjunto, parece que es posible anotar, como también se señalaba en un congreso reciente, que los ingenieros resultan con frecuencia ser demasiado ingenieros (técnicos) y tienden a "ver el mundo por un agujero", alejándose de la visión global de los problemas, de la capacidad de planificar y de la intuición del futuro.

Así pues, los textos adjuntos tienen el aire de querer decirnos que, complementariamente a los principios del nuevo método y al enfoque sistémico y demás técnicas abstractas, una cierta formación humanística ayudaría a combatir el "exceso" de especialización, -no la especialización que de todas formas es ineludible. Entre los textos se intercala alguno que interpreta muy libremente (sustituyendo sólo alguna palabra por otra) párrafos por lo demás literales de Schumacher.

6. Tecnología versus Complejidad

Se recogen en este apartado las que pueden reputarse como las dos caras de la tecnología (aquí ejemplarizada por la informática, tecnología de la información por excelencia) con respecto a la complejidad. Resumiendo mucho, la informática es un instrumento para comprender y manejar la complejidad, pero también genera complejidad.

6.1 El ordenador, instrumento para manejar complejidad

De diversas formas, diversos autores nos indican la potencia del ordenador con vistas a afrontar la complejidad, unas veces para ayudar a gestionar materialmente las instituciones y sistemas sociales del mundo real, otras para modelar un mundo simulado, otras simplemente para profundizar en el conocimiento, ordenando y procesando grandes masas de información.

En cualquier caso, junto a las técnicas de la informática, todos los textos nos manifiestan su insuficiencia si no se acompañan de otras técnicas y métodos: técnicas de sistemas y de cibernética, técnicas de modelación y simulación, técnicas matemáticas de formalización,...

En el mundo real sucede que el aumento de libertad (incluyendo nuevas oportunidades) incrementa la variedad (complejidad) a un punto tal que nuestras respetadas instituciones no son ya capaces de manejarla (...) Sugiero una lista de tres herramientas básicas disponibles para amplificar la variedad de las instituciones: el ordenador, las telecomunicaciones y las técnicas de la ciencia de la organización efectiva (la cibernética) (...) Estoy simplemente proponiendo que la sociedad debería usar estas herramientas para rediseñar sus instituciones, y operarlas de una manera completamente diferente (**Beer**).

El computador, el instrumento de las ciencias de la complejidad, revelará un nuevo cosmos nunca antes percibido. A causa de su capacidad para manejar y procesar enormes cantidades de información en una forma fiable, mecánica, el ordenador, en tanto que herramienta de investigación científica, ya ha revelado un nuevo universo. Este universo era antes inaccesible, no porque fuera muy pequeño o estuviera muy lejos, sino porque era tan complejo que ninguna mente humana podía desentrañarlo (**Pagels**).

¿Podemos entonces eludir las limitaciones al progreso científico impuestas por las limitaciones, cada vez más definitivas, de la observación y de la experimentación? (...) ...aunque la complejidad del mundo real nos impida su observación y experimentación, sí podemos experimentar y observar un mundo simulado. Y para ello disponemos también de una ayuda exosomática, las computadoras. (...) La simulación es, en suma, la gran esperanza para penetrar en la complejidad del mundo (**Wagensberg**).

Complejidad es una evaluación que hacemos de nuestra capacidad para describir, computar o predecir con precisión fenómenos en un dominio (**Denning**).

6.2 Ingeniería, complejidad y economía de medios. El caso del software

Líneas arriba se hablaba de filosofía, aunque fuera en forma entrecorillada. Pues bien, las dos primeras citas que contiene la tabla adjunta se deben a dos filósofos y nos orientan sobre la conveniencia de no introducir en los objetos observados o diseñados más complejidad de la necesaria.

Hasta tal punto es ése un buen consejo que, parafraseando de nuevo muy libremente un texto de Schumacher, podría decirse que "la buena ingeniería es el estudio sistemático de cómo alcanzar metas determinadas con la mínima complejidad general". Únicamente hay que tener en cuenta la pertinencia de considerar la complejidad en el sentido más amplio que permitan las circunstancias, y, a ser posible, en un plano sociotécnico, dado el impacto social de las modernas tecnologías.

Si dos fórmulas de distinta longitud explican un mismo fenómeno con igual mérito, la más corta es verdadera, falsa la otra (**Occam**).

La ciencia debe buscar las explicaciones más simples de los fenómenos más complejos (**Whitehead**).

Los programadores siempre estamos rodeados de complejidad; no podemos evitarlo. Si nuestra herramienta básica, el lenguaje con el que diseñamos y codificamos nuestros programas es también complicado, el lenguaje se convierte en una parte del problema más que en parte de la solución (**Hoare**).

Una regla bastante normalizada es que, si quieres que algo se mantenga simple, la organización que lo desarrolle ha de ser simple (**Warnock**, creador de PostScript).

Todo se debe hacer tan simple como sea posible. Pero para hacerlo hay que dominar la complejidad (**Sampson**, creador de Alto).

Una señal de identidad de los programas muy buenos es que incluso internamente siguen la filosofía de la simplicidad. Si quieren hacer algo complejo llaman al código internamente con operaciones simples, en lugar de hacer la operación compleja desde el principio (**Gates**, fundador de Microsoft).

El verdadero reto es diseñar software que sea simple por fuera, pero complejo por dentro. (...) Yo creo muy firmemente que el diseño y la estructura de alto nivel deben ser hechos por una sola persona, desde luego nunca más de dos. (...) Los programas complicados son mucho más fáciles de escribir que los sencillos y directos. (**Page**, creador de PFS:FILE).

Estudio tras estudio muestran que los diseñadores muy buenos de software producen las estructuras más rápidas, pequeñas, simples, limpias y que requieren menor esfuerzo (**Sackman**).

Si los programas están bien, tienen una especie de simplicidad y de simetría (**Sachs**, creador del programa 123).

La buena ingeniería siempre ha tenido mucho que ver con una economía de medios, con la "usabilidad" (en terminología actual) y con la simplicidad interna del diseño. Sin duda, puntos de vista de programadores de éxito, diseñadores de productos de software reconocidos como triunfadores, pueden aportarnos motivos de reflexión en torno a la complejidad, aunque, como es fácil de constatar, no son mutuamente consistentes.

Y con estos textos finales, si el lector se detiene en compararlos con los del primer apartado, podrá apreciar la profundidad del ámbito de la complejidad, extensible desde un planteamiento global que presenta un panorama prácticamente inabarcable hasta un intento de concreción para los efectos prácticos de un campo tan especializado como es el diseño de programas de ordenador.

7. Bibliografía

Las referencias que se anotan a continuación contienen la mayoría de los textos de donde se han extraído las citas que han servido para componer este capítulo. La bibliografía no ha tratado en ningún momento de ser exhaustiva, como es la tónica general del resto de los capítulos de estas Notas.

Atlan, H.(1979): **Entre le cristal et la fumée**, Seuil, Paris.

Attali, J.(1975): **La parole et l'outil**, P.U.F. Paris.

Beer, S.(1974): **Designing Freedom**, J. Wiley, Londres.

Bunge, M.(1990-91): El sistema técnica-ciencia-filosofía, un triángulo fértil, **Telos**, 24.

Cousins, N.(1979): **Anatomía de una enfermedad**, Kairós, Barcelona.

Denning, P.(1990): Modeling Reality, **American Scientist**, pp.495-498.

Gall, J.(1975): **Systemantics**, Quadrangle, Nueva York.

Gerstner, K.(1988): **Las formas del color**, H. Blume, Madrid.

Illich, I.(1975): **Némesis Médica**, Barral, Barcelona.

Lammers, S.(1988): **Programadores en acción**, Microsoft-Anaya Multimedia, Madrid.

Le Moigne, J.L.(1983): **La théorie du système général**, 2a ed., P.U.F., Paris.

Mélèse, J.(1979): **Approches systémiques des organisations**, Ed. Hommes et Techniques, Suresnes.

Morin, E. (1977, 1980, 1986): **La Méthode, Vols. 1, 2 y 3**, Seuil, Paris.

Pagels, H.R.(1989): **The dreams of reason**, Bantam, Nueva York.

Prigogine, I., I. Stengers.(1983): **La nueva alianza**, Alianza, Madrid.

Racionero, L.(1986): **Arte y ciencia**, Laia, Barcelona.

Rosnay, J.(1977): **El macroscopio**, AC, Madrid.

Rucker, R.(1987): **Mindtools**, Houghton Mifflin, Boston.

Sackman, H.(1968): Experimental studies comparing online and offline programming performance, **C.A.C.M.**, vol. 11, 1.

Sáez-Vacas, F.(1990): **Ofimática compleja**, Fundesco, Madrid.

Schumacher, E.F.(1973): **Small is beautiful**, edición de 1989, Abacus, Londres.

Simon, H.A.(1970): The architecture of complexity, en **The sciences of the artificial**, M.I.T. Press, Mass.

Walliser, B.(1977): **Systemes et modèles**, Seuil, Paris.

Wagensberg, J.(1985): **Ideas sobre la complejidad del mundo**, Tusquets, Barcelona.

Whyte, L.L., A.G. Wilson, D. Wilson.(1973): **Las estructuras jerárquicas**, Alianza, Madrid.

Winner, L.(1987): **La ballena y el reactor**, Gedisa, Barcelona.

Capítulo 10: Un marco universal para la construcción de modelos

En este capítulo consagramos el concepto de "situación compleja", que, a nuestro juicio, tiene la capacidad de integrar coherentemente las muy diversas concepciones de "complejidad". Cualquier situación viene definida en todo momento por la interacción de valores particulares extraídos de tres conjuntos a los que llamamos H (observador, generalmente un ser Humano), I (Interfaz, Instrumental) y O (Objeto). Al resultado de la interacción se le denomina IO (Imagen del Objeto).

Probablemente, no hay nada en ese modelo que no haya sido estudiado en una u otra forma por la filosofía, en particular por la epistemología, pero a él se ha llegado en este caso desde un planteamiento técnico orientado a sintetizar una herramienta muy condensada, sencilla de recordar, y práctica para comprender y afrontar un número ilimitado de situaciones reales envueltas en complejidad.

Pero en realidad lo que veremos es, más que un modelo, un marco para construir modelos y por tanto su alcance desborda aparentemente el objetivo inicial, que lo relacionaba con la complejidad. Si decimos "aparentemente" es porque, en definitiva, modelar es siempre una manera de "manejar complejidad".



1. Introducción

Este capítulo trata de un modelo general que da cohesión y sentido a todo este texto. Si lo hemos situado en medio y no al principio del mismo ha sido por una decisión muy meditada. Queríamos respetar el orden epistemológico personal, es decir, la génesis del modelo, que ha surgido dentro del autor después de varios años de pelear con cuestiones de sistemas y de complejidad. Por analogía, los capítulos anteriores deberían actuar sobre el lector como preparación psicológica del terreno para poder asumir mejor este modelo como un planteamiento global de todo lo que ya ha visto.

La distribución del capítulo es sencilla. Primero, veremos el marco para la construcción de modelos, a través de una exposición del significado de las letras H, I, O e IO y de su manejo constructivo. A continuación, abordaremos su aplicación, que hace emerger el concepto de "situación compleja o de complejidad", con el que reencontraremos varios de los tipos de complejidad ya visitados en capítulos anteriores o sugeriremos algunos nuevos que serán tratados en capítulos posteriores.

Dedicaremos una atención especial a la letra I, por su impacto decisivo en el filtraje de la complejidad del objeto, y por tanto en su construcción mental. Habitualmente, la I engloba todo el arsenal teórico, conceptual o instrumental a disposición del observador. La teoría de sistemas o el enfoque sistémico son elementos del conjunto I, por ejemplo. Por un razonamiento recursivo, incluso el modelo $H \times I \times O$ es otro elemento del mismo conjunto.

Para terminar, se harán algunas anotaciones acerca de la capacidad del modelo para absorber los variadísimos conceptos que destilan los textos sobre complejidad seleccionados en el capítulo anterior.

2. Los elementos

A estas alturas del texto, puede que el lector tenga una idea bastante aproximada de lo que es la complejidad, sobre todo si ha trabajado personalmente las referencias bibliográficas o si ha debatido en clase las cuestiones. Pero también es posible que esté hecho un lío después de tantas clasificaciones, aproximaciones, enfoques y definiciones.

Aquí intentaremos encajar en un sencillo esquema conceptual la mayoría de las ideas que sobre complejidad han aparecido hasta el momento.

El punto de partida es un hecho bien conocido: nuestra mente es limitada. Carece de la capacidad para aprehender todos los hechos que ocurren en el "mundo real", todos los elementos e interrelaciones. No es capaz de manejar directamente la complejidad. Para ser más exactos, ni siquiera lo intenta.

De forma más o menos consciente, todos filtramos la realidad y afrontamos una complejidad drásticamente reducida. Ciertamente, en lugar de tratar directamente con la realidad -entendida como el conjunto completo de factores que determinan un objeto- construimos modelos de ella. El proceso mediante el cual se obtienen estos modelos y las implicaciones sobre reducción de complejidad que de él se desprenden, constituyen la materia de este capítulo.

2.1 O: el objeto

Cuando contemplamos el mundo, sólo percibimos una parte de él. Centramos nuestros sentidos, nuestra atención o nuestro estudio en algo, en un **objeto**, al que nos referimos por un nombre. De la infinidad de componentes del universo, elegimos un subconjunto: digamos que definimos las "fronteras" de lo que nos interesa. Pueden delimitar éstas lo que llamaremos una piedra, un ordenador, una institución o una célula. Y esta delimitación no es únicamente espacial: también fijamos la realidad en el tiempo, centrándonos en su estudio en un intervalo temporal limitado.

Parece claro que la complejidad que hemos de manejar disminuye cuando pasamos de considerar el "mundo real" en su totalidad a ocuparnos sólo de uno de sus fragmentos. Esta disminución puede entenderse si tenemos en cuenta la drástica reducción en el número de elementos y relaciones consideradas. Pero en cualquier caso, el nivel de complejidad, pese a haber disminuido mucho respecto al universo entero, puede seguir siendo grande si continuamos manejando un número elevado de interrelaciones.

Veámoslo con un sencillo ejemplo. En él, el *O* elegido es una piedra concreta. En términos, por ejemplo, de partículas, hemos pasado de considerar todas las del universo a tener en cuenta únicamente las que forman la piedra. Pero aún así, nos queda un ingente número de ellas (la piedra estará compuesta por una grandísima cantidad de átomos, por ejemplo).

2.2 I: la interfaz

Cuando nos ocupamos del objeto que hemos elegido para nuestro estudio, lo hacemos de una cierta manera. Utilizamos siempre, de una forma más o menos explícita, unos instrumentos y unas técnicas. Estos instrumentos y técnicas constituyen lo que Sáez Vacas llama **interfaz** o instrumento. Es evidente que aquí la voz "interfaz" se utiliza en el sentido más general posible, que incluye por supuesto a la interfaz tecnológica de manejo de un aparato.

En el caso más directo, percibimos la realidad a través de nuestros propios sentidos, que ya por su propia naturaleza realizan un importante filtrado de complejidad y nos focalizan sobre ciertas características del objeto. Y utilizamos unas técnicas, como por ejemplo las que comprende lo que solemos llamar "sentido común". (Por razones que se explicarán más adelante, los sentidos, y en general el aparato perceptivo, los englobaremos como un componente del factor H).

Podemos usar, también, un microscopio. Así centraremos nuestra atención de forma que todo lo que sea mayor de un cierto tamaño nos resultará irrelevante, simplemente no lo veremos. Lo mismo pasará con lo que sea menor que la capacidad de resolución del aparato. Con un microscopio electrónico pueden verse las cadenas moleculares, mientras que para ver las estructuras celulares, que son construcciones arquitectónicas de cadenas moleculares, es suficiente con un microscopio óptico; por su parte, las células se pueden estudiar con un microscopio óptico y, a veces, con una lupa. Sin embargo, si lo que utilizamos es un telescopio, la realidad cercana desaparece, y nos centraremos en los cuerpos que queden dentro de su alcance.

Veamos otro ejemplo. Si estudiamos un ordenador con las herramientas conceptuales que nos proporciona la electrónica, sólo podremos ocuparnos de lo que podríamos llamar "nivel electrónico", y no percibiremos el nivel lógico, ni por

supuesto nada que tenga que ver con los procesos que en él se realizan y que se organizan por las técnicas de la programación.

En resumen, con un instrumento vemos una cosa pero dejamos de ver otras. Si vemos la molécula, no vemos la luna. Y viceversa. Todos los instrumentos producen simultáneamente un efecto amplificador y limitador.

De estos ejemplos parece desprenderse la idea de que hay dos tipos de interfaz (cuya naturaleza discutiremos con posterioridad):

- a. Por un lado, lo que podríamos llamar "**herramientas**": métodos o instrumentos físicos utilizados en la percepción del objeto considerado.
- b. Por otro, lo que serían "**técnicas**": métodos conceptuales de análisis de los datos que nos han aportado los instrumentos o, simplemente, sistemas de representación. En un sentido amplio, todos éstos son también **herramientas cognitivas**.

En cualquier caso, ambos tipos de interfaz están muy relacionados: según las características de los datos que hayamos obtenido, tendrá sentido utilizar unas técnicas y no otras. Incluso, en bastantes ocasiones la técnica viene "impuesta" por un determinado tipo de datos, con lo que la herramienta queda definida dentro de un estrecho margen.

La elección del instrumento condiciona la observación del objeto, su complejidad y por tanto su solución, si se trata de un problema. El observador tiene ante sí la difícil alternativa de distribuir complejidad entre dos dimensiones ortogonales, la dimensión de profundidad y la de amplitud de campo, partiendo naturalmente de la base de que hay un insoslayable fenómeno de reducción de variedad en comparación con la que percibiría un hipotético "observador omnipotente". (Nota: el lector debe tener muy claro que cuando se habla de variedad es habitual manejar dos puntos de referencia en cuanto a la categoría del observador, el observador omnipotente y el observador a secas, o sea, nosotros. Con respecto al segundo, un instrumento adecuado puede amplificarle la variedad del objeto observado. Si no fuera así, no tendrían sentido los instrumentos tecnológicos ni los sistemas de enseñanza).

Amplitud de campo significa más elementos, con menos detalles. Por ejemplo, eligiendo el microscopio, no sólo se desdeñan las características que quedan fuera de su amplitud de campo, sino también todas las que no producen un efecto luminoso dentro de las longitudes de onda para las cuales las lentes son transparentes. A cambio, este instrumento revela muchos detalles de una pequeña porción del objeto. Profundidad igual a menos elementos, con más detalles. Un voltímetro de precisión aplicado sobre un circuito eléctrico nos proporciona certeramente un

número entre muchos posibles, pero es ciego a una gran variedad de fenómenos o propiedades, como forma, temperatura, interacciones electrónicas, propiedades químicas, etc.

Desde el punto de vista de la teoría de la variedad, podemos considerar a todo I como un filtro de variedad, colocado entre el objeto y la mente humana.

2.3 H: el observador

Y por último, tenemos a la **persona**¹ que está analizando (o haciendo cualquier otra operación, a la que genéricamente llamamos "observación") el objeto. Y la tenemos no como un "ente neutro", sino como un ser pensante, con su historia, su forma de ser, sus facultades perceptivas, sus conocimientos, sus opiniones y sus intereses: no es lo mismo observar una piedra para desentrañar su composición que para lanzarla. Utilizamos la letra H como símbolo recordatorio de que generalmente el observador es un ser **humano**.

De nuevo tenemos un filtraje de complejidad o de variedad: entre todos los aspectos del objeto accesibles a través de la interfaz, el observador sólo capta o toma en cuenta aquéllos que son conformes a su capacidad o intereses. En cualquier caso, el conjunto de sus circunstancias personales le impiden ser "objetivo" y considerar todos los datos por igual y ello al margen de las características de la interfaz. Dos observadores, con los mismos instrumentos, y usando las mismas técnicas, pueden obtener visiones o al menos interpretaciones radicalmente distintas de un mismo problema, en especial si éste involucra aspectos humanos.

Un caso particularmente interesante para nosotros sucede cuando el propio H es quien elige el objeto y la interfaz entre un conjunto de posibilidades, de acuerdo con sus intereses y necesidades. Tal elección es fundamental, puesto que en ella se juega el diseño del filtro de complejidad, esto es de la clase y cantidad de información que va a contar en el proceso de observación.

De los elementos vistos hasta el momento (H,I,O), es H con seguridad es el que de forma más explícita e insistente nos hemos encontrado en otros capítulos y siempre le hemos llamado **observador**. Así, cuando estudiamos la variedad, llegamos a la conclusión de que ésta era una propiedad del binomio objeto-observador: recuérdese cómo, en el ejemplo del neurofisiólogo y el carnicero contemplando un cerebro -"sesos" para el segundo-, cada uno percibía una variedad diferente. En el capítulo sobre **marcos conceptuales**, las definiciones de complejidad debidas a G. Klir, F. Sáez Vacas o R. Flood integraban al observador. Este último desagrega

¹En sentido amplio, el observador no ha de ser necesariamente una persona: el marco permite también que sea, por ejemplo, una máquina quien se enfrente al objeto.

la complejidad en atributos relacionados con el observador y en atributos relacionados con el objeto. Y, para redondear, el capítulo sobre **Textos** contiene una sección muy explícita acerca de la "subjetividad en la complejidad".

2.4 IO: La imagen del objeto

Todos los filtrajes de variedad (en un sentido absoluto: reducciones de variedad), nos dan al final un objeto modelizado, idealizado, de una complejidad incomparablemente menor que la del "universo real". Con esta complejidad es con la que se enfrenta nuestra mente. Lo que aquí llamamos modelo es un concepto muy parecido al de sistema, según la concepción que de "sistema" nos ha dejado primero Ashby, y después se ha encargado Klir de reforzar: cuando se establece un conjunto de variables como resultado de nuestra interacción con el objeto que nos interesa, entonces se dice que se distingue un sistema en el objeto. Un sistema es la abstracción creada a partir de un objeto real, una **imagen**.

En el **modelo** (o **Imagen del Objeto**, de ahí lo de IO), los elementos son muchos menos, y las relaciones entre ellos mucho más sencillas que en la realidad. Naturalmente, para llegar hasta aquí hemos perdido una cantidad importante de información. Pero como contrapartida, nos enfrentamos con algo mucho más sencillo de comprender. Así, lo que varía de una situación a otra, de un problema a otro, no es la complejidad propia de la cuestión que estemos tratando, sino la del modelo del universo que hemos creado para ese caso particular. En muchos casos podría decirse que cuando se considera a algo como muy complejo, lo que ocurre es que no se ha encontrado un modelo de ello que sea suficientemente sencillo.

Conviene advertir una vez más, no obstante, que decir que la modelización supone una reducción de variedad por referencia a la variedad generalmente inabarcable del objeto no es contradictorio con el hecho de que el progreso tecnológico, al aumentar el poder de las herramientas o interfaces, amplifica ante los observadores, para bien y para mal, la variedad perceptible del mundo (véase al respecto más adelante el capítulo dedicado a las **tecnologías de la información**). Precisamente, esta última circunstancia reclama a gritos la necesidad cada día más urgente y la dificultad de manejar todo ese flujo de variedad que recorre la tarea humana, la necesidad de crear y distribuir macroscopios.

3. U.F.M.: un marco universal para la construcción de modelos

Ya hemos introducido todos los elementos del marco conceptual que propone Sáez Vacas. Después de haber tomado contacto con ellos uno por uno, nos adentraremos en hacernos una idea de cómo se relacionan entre sí.

Hemos dicho que siempre que se intenta manejar la complejidad del mundo, aunque no sea más que simplemente para comprenderlo, se produce un "**proceso de modelación**", que representamos mediante el siguiente esquema-fórmula:

$$\mathbf{H \times I \times O \rightarrow IO}$$

Podemos entenderlo como un modelo de adquisición de conocimiento, de percepción de la realidad: un modelo para construir modelos. En definitiva, como la forma en que manejamos un mundo que es, por definición, extremadamente complejo, con una mente que sólo puede abarcar parte de esa complejidad. Por su total generalidad y máximo grado de abstracción bien la podemos considerar como un marco universal para la construcción de modelos, de ahí sus siglas U.F.M. (**U**niversal **F**ramework for **M**odeling, en inglés).

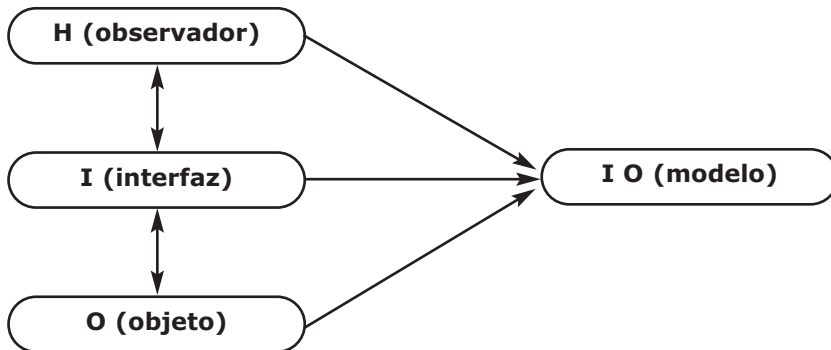


Fig. 1. Los elementos del marco $\mathbf{H \times I \times O \rightarrow IO}$ (F. Sáez Vacas)

3.1 El proceso

En el esquema (o fórmula), aparecen todos los elementos que hemos venido presentando anteriormente. Pero, además, aparecen relacionados de una determinada forma, similar a la expresión matemática del producto de conjuntos. En una primera aproximación, podría leerse "La interacción entre un observador, una interfaz y un objeto determinados produce un modelo del objeto". Por tanto, la complejidad que podamos observar en ese modelo (imagen del objeto) será la

consecuencia de la interacción de los tres elementos del primer miembro de la fórmula.

Siempre que nos acercamos a un problema, delimitamos una parcela de la realidad² (el objeto, O), usamos ciertas técnicas e instrumentos (en nuestro esquema, el I), con unos determinados objetivos, intereses y motivaciones (los del ser humano, H). Y así "creamos" nuestra propia visión del asunto. Sobre ella resolvemos el problema. Y luego extrapolamos de nuevo sobre la realidad.

Aquí es donde se ve si la reducción (a veces simplemente elección) de complejidad es la apropiada para nuestros fines, o si hemos despreciado alguna relación o elemento que deberíamos haber tenido en cuenta. Como toda reducción de complejidad implica pérdida de información, puede ser que hayamos eliminado en el proceso algo que es determinante para lo que buscamos. De ahí la gran importancia de haber elegido bien el objeto y la interfaz.

Conviene decir alguna palabra más sobre el concepto de interfaz. Antes hemos hablado de dos grandes tipos: interfaces o herramientas físicas y herramientas cognitivas. En una primera aproximación, las primeras se aplican directamente al mundo real y producen un "salto" del mundo físico al de los modelos, al de las ideas: el I "transforma" el objeto real en un conjunto de datos. En tales casos, la interfaz suele ser ella misma un objeto real (sentidos de una persona, microscopio, sensor de temperatura, un computador,...). Hay que hacer notar que el microscopio, por sí mismo, no produce el modelo. Lo que "sale" por su objetivo no es más que luz. Será en la mente del observador donde se genere la "imagen del objeto", entendida como modelo.

Y por otro lado están las interfaces que actúan sobre modelos, producto de un proceso de modelación anterior. Ahora no hay "cambio de mundo", aunque sí se produce un modelo (IO) menos complejo. La reducción de complejidad viene dada por el desprecio de relaciones y datos, lo que se puede hacer de una forma más o menos acertada. Además, estas interfaces son ellas mismas de naturaleza "intangibles": son modelos de acercamiento al conocimiento (en general, IOs producidos por otros procesos de modelización). A estas interfaces las llamamos técnicas, métodos, teorías, modelos, etc. El software de ordenadores entra más en esta categoría que en la primera.

²En un sentido más amplio, deberíamos decir que delimitamos una parcela del universo de discurso ya que, como veremos, O puede ser un objeto intelectual (por ejemplo, un modelo).

Finalmente, ambas categorías pueden combinarse, y de hecho es lo que sucede mayormente en la práctica actual, en la que los instrumentos, cada día más sofisticados, están controlados por ordenador, cuyo software a su vez es manipulado por el usuario por medio de una interfaz (en su acepción informática) conveniente.

Veamos como ejemplo el caso de un biólogo estudiando una célula.

En una primera etapa, tenemos un proceso de modelación donde el objeto es la célula, el observador es el biólogo, y la interfaz, el conjunto formado por el sistema de visión del biólogo y el microscopio que está usando. Este instrumento material permite la elaboración de un modelo a partir de un objeto real. En este caso el modelo sería una "imagen" de la célula en el cerebro del investigador.

Pero no es esta imagen con la que trabaja nuestro biólogo. De una forma más o menos consciente, tiene lugar un segundo proceso de modelación: ahora el O es la imagen de la célula, el H sigue siendo el mismo, y el I son las técnicas de identificación de estructuras celulares, que usa para identificar la célula en términos de los elementos que la forman (membranas, citoplasma, ribosomas, etc.). La elaboración del modelo ha partido ahora de otro modelo: para ello hemos utilizado (ha utilizado el biólogo) una técnica. Insistiremos sobre este proceder.

3.2 Significación "dinámica" de H, I, O

Aquí, nos vamos a ocupar de ilustrar al lector sobre algunas de las múltiples posibilidades interactivas, que no son sólo las que se evidencian a primera vista, estáticamente, sino las que pueden secuenciarse jugando a conveniencia con los términos de nuestra fórmula. Esta es la razón para utilizar el adjetivo "dinámico".

Hasta ahora hemos considerado únicamente objetos, interfaces y personas, "sin más adornos". Pero se puede ampliar el sentido del proceso de modelación descrito si consideramos que *H*, *I* y *O* pueden ser a su vez modelos (*IO*), o conjuntos de modelos, producidos por otro proceso previo de modelación ($H \times I \times O \rightarrow IO$). Podríamos estudiar desde este punto de vista la jerarquía de sistemas propuesta por Klir (ver **Marcos Conceptuales**): cada nuevo nivel epistemológico aparecería al considerar el sistema (*IO*) del nivel anterior como objeto (*O*), y aplicar sobre él un proceso de modelación.

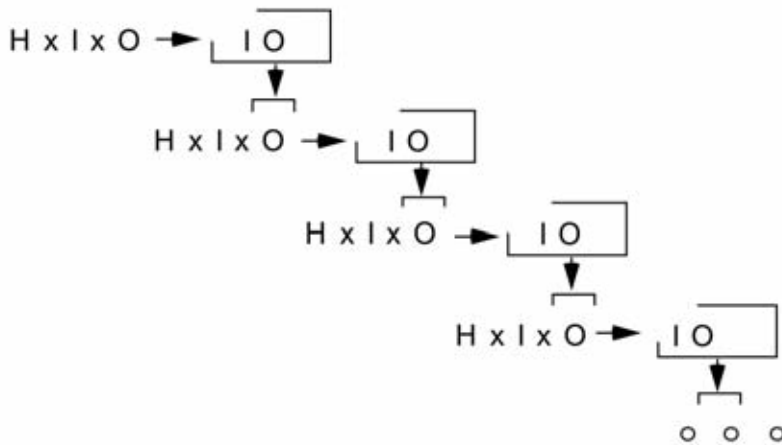


Fig. 2. Cadena de procesos de modelado (donde el O de cada uno es el IO del anterior)

De esta manera, O será, bien el resultado de una parcelación de la realidad, bien un IO o conjunto de IO s producidos en otros procesos previos. En este último caso, estaremos ante un "modelo de modelo", esto es, una mayor reducción de la complejidad respecto a la de la realidad con que tratamos. Es lo que ocurre, por ejemplo, cuando estudiamos un problema típico de mecánica: primero modelamos cada uno de los objetos y relaciones que intervienen en él (obtenemos así "puntos materiales", "fuerzas", etc.), y luego, tomando el conjunto de estos modelos como O (que sería el enunciado del problema), los relacionamos según nuestros conocimientos, y obtenemos el IO , que es el modelo del problema que utilizamos para resolverlo³.

³Otro ejemplo que puede ayudarnos es el del estudio de un sistema electrónico (como una CPU). Comenzamos por considerar los elementos reales (cables, resistencias, transistores, chips, placas de circuitos,...) como objeto. Los analizamos mediante las herramientas que nos proporciona la teoría de circuitos y obtenemos una imagen del objeto: el esquema del circuito. En un nuevo proceso de modelado, ese esquema será el objeto. Otro observador (especializado en electrónica digital) analizará ese circuito y obtendrá a partir de él una nueva imagen del objeto: un diagrama de puertas lógicas, por ejemplo. Así se ha llegado a algo que podríamos llamar "la imagen de la imagen del objeto".

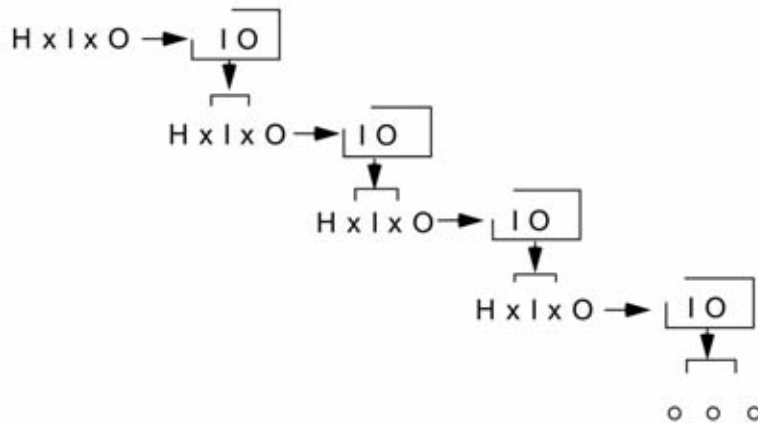


Fig. 3. Cadena de procesos de modelado (donde el I de cada uno es el IO del anterior)

De igual forma, podemos usar un *IO* como técnica para disminuir la complejidad de un objeto. En este caso, actuaría como interfaz en el proceso de modelado correspondiente. Podemos encontrar un ejemplo electrónico para esta posibilidad. Considerando como *O* un variado conjunto de circuitos electrónicos, y usando como *I* el método científico, puede deducirse la teoría de circuitos (*IO*). A través de un nuevo proceso de modelación podremos usar esta teoría como *I* aplicada ahora a un circuito concreto (el nuevo objeto), obteniendo como imagen del objeto su imagen circuital.

La recursividad del proceso alcanza también al observador (*H*), que muy bien podría ser un modelo (*IO*) de un observador real. Esta situación es muy corriente. Cada vez que uno de nosotros utiliza un método de trabajo en una empresa, esta manejando un instrumento (*I*) que alguien, otro *H*, ha diseñado o elegido anteriormente pensando en un observador abstracto promedio al que nos asemejamos por conocimientos y características cognitivas.

Supongamos un caso habitual en la profesión de ingeniero: un proyecto de diseño y construcción de software. El equipo humano formado para tal cometido es un conjunto de *Hs* con un *H* responsable al frente, al que se llama jefe del proyecto. La tarea básica de este último, consistente en distribuir y coordinar el trabajo de los demás, se concreta en definir un conjunto adecuado de subesquemas $H(i) \times I(i) \times O(i)$, un subesquema para cada miembro o subconjunto de miembros del equipo, sean éstos diseñadores, analistas, programadores, documentalistas, técnicos de sistemas, etc. El jefe del proyecto se fabrica una imagen (*IO*) del software, la fragmenta en subimágenes (*IO(i)*) -subsistemas, etapas o lo que sea-, que entrega como objetos (*O*) a los distintos miembros del equipo, $H(i)$, junto con

la definición de la técnica de trabajo (I) a utilizar por éstos -método de análisis, lenguaje de programación, instrumental informático, manual de procedimiento documental, controles de calidad, planificación-. De esta manera tan condensada, queda descrito el amplio espectro de tareas, saberes y experiencias que hipotéticamente habría de abarcar un jefe de proyecto de algo tan complejo como es un proyecto de software, tan amplio que incluso se extiende de manera predominante a la modelación de los seres humanos.

Con esta inmersión en las diversas posibilidades significativas del esquema U.F.M, ya podemos postular que lo que venimos llamando "proceso de modelación" es un marco suficientemente vasto como para que encaje en él cualquier forma de acceso a la realidad, y que supera el objetivo propuesto de manejar la complejidad.

4. Aplicación del U.F.M. al estudio de la complejidad

A lo largo de estas notas nos hemos encontrado y nos encontraremos aún con diversos tipos y definiciones de complejidad, lo que crea una sensación enojosa de falta de coherencia, consistente en que cada cual maneja "su" noción de complejidad. Es evidente que carecemos de una teoría unificada al respecto, pero como aquí tratamos de construir una **arquitectura socio-técnica de las tecnologías de la información tomando como hilo conductor sus dimensiones de complejidad**, necesitábamos un concepto que acogiera y diera sentido diferenciado a todos los tipos de complejidad. Dicho concepto es el esquema $H \times I \times O$. En otras palabras, cualquier tipo o definición de complejidad se enmarcaría en un caso particular de valores de los términos de este esquema.

Sólo por dar un ejemplo, capítulos atrás (**Conceptos relacionados con la complejidad**) se establecía una distinción entre complicación y complejidad. La complicación de un algoritmo se medía por el número de instrucciones necesarias para describirlo en un lenguaje específico, mientras que la complejidad de ese mismo algoritmo podía medirse por el tiempo necesario para su ejecución en una determinada clase de máquina. Según nuestro enfoque, complicación y complejidad de un algoritmo son dos aspectos particulares de complejidad relativas a un mismo objeto, el algoritmo.

En el primer caso, sucede que un observador -caso particular de H- está interesado en un cierto tipo de dificultad con relación al algoritmo, el objeto O. Las herramientas de observación o de medida I que aquél aplica, movido por su interés y profesión, son un lenguaje de programación (máquina virtual específica) y alguna métrica específica, quizá técnicas derivadas de la ciencia del software, basadas en el número de operadores y operandos de Halstead (véase más adelante capítulo sobre **Medidas de la complejidad del software**). Resumiendo, la imagen IO que el programador obtiene es un conjunto de valores proporcionados por esa métrica concreta.

El segundo caso se suscita ante otro tipo de observador y de situación, muy frecuente en el campo del diseño de algoritmos y computadores paralelos, y cuyo interés se concentra en desplazar los límites físicos de la computación. El observador suele ser un avezado especialista que, aplicando sobre el algoritmo ciertas técnicas matemáticas estimativas del volumen de operaciones elementales necesarias, proyecta una imagen del algoritmo resumida en una expresión formal (en casos sencillos, función del tamaño de los datos de entrada al algoritmo).

Como se ha podido ver, un mismo objeto provoca dos situaciones bien distintas. Detrás de una de ellas puede haber motivaciones relativas a la dificultad de programación y toda su secuela de problemas de coste del software a lo largo de su ciclo de vida. Detrás de la otra, se esconde el deseo de computar cada vez más deprisa y más. Sin embargo, ambas son interpretables por el mismo esquema utilizado sistemáticamente.

4.1. Situaciones de complejidad y sus factores componentes

En cada situación emerge o puede emerger una circunstancia genuina generadora de complejidad, con objetos, observadores e instrumental específicos. Vamos a llamarla, para entendernos, situación de complejidad.

Tal noción eliminaría parte del confucionismo reinante, del que, como muestra ilustre -por su autor- citamos el libro de H.R. Pagels, referenciado en la Presentación de estas Notas y profusamente extractado en el capítulo sobre **Textos**. Ahora nos referiremos a su capítulo tercero, titulado Orden, Complejidad, y Caos. Pagels empieza preguntándose ¿qué es complejidad?, para declarar inmediatamente después la necesidad de ir hacia una definición fuerte, hacia una medida cuantitativa, "que pueda ser asignada a un sistema físico o a una computación, y que se sitúe entre la medida del orden simple y del caos completo".

Examina varios conceptos (tipos, decíamos antes) de complejidad: complejidad algorítmica, complejidad computacional, complejidad basada en la información, complejidad física y profundidad lógica. Sin embargo, no llega a ninguna conclusión. Cualquier lector de estas Notas podría ahora describir sin vacilar esos cinco conceptos como cinco situaciones concretas de complejidad, denotando sus valores H, I, O, e IO.

Con respecto al trabajo de identificación práctica de estos valores, daremos a continuación un bosquejo de algunos de los más corrientes.

intereses	objetivos/propósitos	implicaciones
actividades	funciones	efectos
factores cognoscitivos	factores perceptivos	factores ideológicos
modelos epistemológicos	conocimientos	estructura organizativa
.....

Tabla 1: Componentes de H (observador Humano)

Con respecto a un objeto un observador puede tener diversos intereses, objetivos o propósitos. Pero también producen imágenes distintas de un objeto sus implicaciones o efectos sobre el observador (un sistema informático tiende a ser visto de forma bien diferente por el director de una agencia bancaria y por el empleado que se ve obligado a convertirse en operador), y las actividades y funciones que con respecto al objeto tienen que realizar los distintos observadores. De forma parecida actúan los factores cognoscitivos y otros factores muy profundamente arraigados en la persona individual, como pueden ser los factores ideológicos, si éstos tuvieran alguna relación con el ámbito de propiedades del objeto considerado. Un ejemplo puede ser la condición de marxista militante frente al problema del diseño social. Pero no debe el lector concluir a partir de este ejemplo que los factores ideológicos se asocian necesariamente a planteamientos políticos, económicos o religiosos, sino que son asociables, y mucho, a los campos científico y técnico. En el campo técnico, el autor de estas Notas ha detectado y definido cinco subculturas informáticas (véanse referencias bibliográficas en la **Presentación**), que impregnan de forma muy diferenciada la actuación de sus respectivos "militantes" frente a un mismo "objeto" informático.

Muy especialmente influyen los factores perceptivos: la percepción es no sólo limitada ("si se abriesen las puertas de la percepción, cada cosa aparecería al hombre como es: infinita", escribió William Blake), sino muy distinta de una persona a otra.

¿Quién puede dudar de que nuestros conocimientos filtran absolutamente la realidad que percibimos? Un ingeniero de telecomunicación, un empresario y un miembro del servicio de espionaje, puestos a redactar un informe sobre lo que "ven" en un sistema telefónico, diferirán bastante, y no exclusivamente por sus intereses y actividades peculiares. En cuanto a la diferenciación de imágenes generadas por el modelo epistemológico aplicado, recordemos el ejemplo, expuesto en el capítulo anterior, de las cinco formas matemáticas de describir la mano humana. De modo más general, los filósofos han acuñado el término de "Weltanschauung" para denominar al conjunto de factores culturales profundos que en un momento histórico determinado configuran la visión del mundo de una colectividad. La "Weltanschauung" es un modelo epistemológico diferenciador de colectividades,

tanto en el tiempo como en el espacio. En la época de Newton el universo estaba gobernado por una sola fuerza, la gravitatoria, mientras que hoy es comúnmente aceptada entre los físicos la existencia de cuatro fuerzas: la gravitatoria, la electromagnética, la débil y la fuerte.

Podrían considerarse otros muchos factores, aunque sería inadecuado extenderse más en estas Notas, y, además, hay que dejarle al lector la iniciativa de intentarlo por su cuenta. Le sugerimos que anote que, de una u otra forma, todos ellos podrán ser clasificados aproximadamente en dos **grupos**: el de relaciones del observador con el objeto o el grupo de sus posibilidades discriminatorias **personales**.

Lo que sí interesa resaltar, antes de acabar con el término H, es que éste, aparte de poder ser un objeto artificial (caso cuyo tratamiento estamos ignorando), también puede ser un grupo de personas, unido por una estructura organizativa. Este caso es muy interesante, ya que, siendo aplicable a entidades tales como un grupo de trabajo, el conjunto de una empresa o un colectivo ciudadano, su impacto en la observación de un mismo objeto se desglosa en una multiplicidad de imágenes: una fuente muy habitual de complejidad (y de dificultades).

instrumental	interfaz	método
sistema de representación	norma operativa	teoría
ideología	modelo	cultural empresarial
....

Tabla 2: Componentes de I (Interfaz)

Asimismo, el término I puede representar una variedad de valores, de los que en el cuadro adjunto explicitamos algunos a título de ejemplos. Haremos muy pocos comentarios.

De forma habitual, y por razones de oficio, de afición u otros, el factor I va asociado al H concreto. Pero en todo caso involucra **componentes** que le son externos o relativamente **externos**, en el sentido de que no están integrados profundamente en su personalidad y quehacer, y por tanto tienen posibilidad de ser alterados, es decir, cambiados, complementados, elegidos o adecuados. Esta es una frontera un tanto difusa entre los términos H e I, sobre la que es preciso que el lector reflexione, para que pueda distinguir qué ideologías, modelos o teorías son clasificables a un lado u otro de la misma.

El modelo HxIxO entra en la categoría I. Más exactamente, es un instrumento epistemológico muy general con el que un observador H1 observa un objeto constituido por otro(s) observador(es) H2 observando mediante un I2 a un objeto O2. Y recuérdese que "observar" es el nombre que empleamos genéricamente para observar, diseñar, construir, usar, etcétera. Hemos visto un buen ejemplo de este mecanismo recurrente de observación de segundo orden cuando describíamos el trabajo del jefe de un proyecto de software, pero ahora, después de desvelar estos cuadros de componentes, el lector tiene más posibilidades de concretar nuestra mención de entonces al "amplio espectro de tareas, saberes y experiencias" convenientes para realizar ese trabajo.

Como extensión de lo que se dijo en apartados anteriores dedicados a I, introduciremos ahora una consideración acerca de la significación amplia que estamos queriendo atribuir a epígrafes como el de "sistema de representación". Con él designamos tanto un nivel descriptivo de computadores, como un lenguaje de programación, una formulación matemática o un sistema de proyección geométrica, pero también una campaña de propaganda, una conferencia académica o una clase magistral en la universidad. Todas estas instancias filtran la realidad, de la que capturan una imagen más o menos conveniente, interesante o útil, según los puntos de vista. Si repasamos el ejemplo en el que tomábamos en cuenta la posibilidad de que H sea un colectivo, la actividad de definir, construir y difundir un determinado conjunto de ideas e informaciones, tomada como un sistema de representación, puede jugar un papel consistente en reducir la multiplicidad de imágenes en el colectivo, y por tanto la complejidad de manejo de su estructura organizativa.

El pedazo de realidad al que hemos llamado **O** se proyecta por la interacción con **H** e **I** en apariencias muy diversas, de las que en el cuadro adjunto recogemos algunas muestras.

ente	proceso	actividad
área de conocimiento	área de actividad	SISTEMA
relación	conjunto	concepto
imagen gráfica

Tabla 3: Formas de IO (Imagen del Objeto)

Así, cuando miramos un objeto con el enfoque sistémico la imagen que de él producimos es un sistema.

La imagen gráfica es un ejemplo trivial de imagen de un objeto real, pero también puede ser un caso especial cuando se trata de la generación de formas computadas. Ahí, el objeto (O) es un conjunto de números e I es el algoritmo y el grupo de técnicas que los organizan, les dan significado y los proyectan de alguna forma gráfica (IO) sobre una pantalla o un papel.

4.2 Construcción de marcos específicos

Hemos hablado de la frontera difusa entre H e I. Otra cuestión a subrayar es que, además, H e I son siempre **elementos multidimensionales** o multivaluados, lo que quiere decir que cualquier situación viene caracterizada no por uno sino por varios (a veces, muchos) valores a la vez de los componentes reflejados en los cuadros anteriores.

Con frecuencia, dichos **componentes** pueden ser también **interdependientes**, como muestra la siguiente observación de D. Bohn y D. Peat en su libro "Ciencia, orden y creatividad". Los experimentos demuestran que el flujo de información procedente de niveles más elevados del cerebro a las áreas de construcción de imágenes visuales supera la cantidad de información que llega de los ojos. Dicho en otra forma, lo que "vemos" es tanto producto de un conocimiento previo como de la acción fisiológica específica. "Así pues, la percepción sensitiva se halla fuertemente determinada por la disposición global de la mente y del cuerpo. Pero, a su vez, esta disposición se relaciona de manera significativa con la totalidad de la cultura y de la estructura social".

Generalmente, todo eso es lo que hace compleja a una situación, y, precisamente, en seleccionar una o pocas dimensiones adecuadas para obtener una IO razonable es en lo que consiste mayormente la operación de simplificación (releer ahora capítulo sobre Simplificación), lo mismo que la complejificación planificada podría consistir en introducir dimensiones o valores no considerados inicialmente o en afinar el aparato de observación ampliando el ancho de banda.

Fijando, eligiendo o desechando valores de los términos H, I y O es también como se construyen los marcos de estudio o de acción sobre los objetos. El único problema es que estas operaciones no se suelen explicitar.

Así, por ejemplo, S. Beer, un H concreto, cuyas características personales no tienen por qué conocer sus lectores, en su libro "Designing Freedom" considera ciertos objetos llamados instituciones sociales, a los que ve como sistemas, puesto que él es un destacado sistemista. Aplicando este enfoque, los caracteriza de una manera típica como sistemas inestables, incapaces de dar servicio al ciudadano. Una vez explicado este diagnóstico como si fuera una verdad absoluta, y no una "situación", diseña otra situación en la que él ya no interviene, sino que lo harán unos Hs abstractos (los componentes humanos que controlarán dichas instituciones), desprovistos de

cualquier otra dimensión, quienes, aplicando un instrumental I, formado por las nuevas tecnologías de la información y por la herramienta intelectual cibernética, conducirán a la institución (es decir, a su imagen, el sistema), primero a la estabilidad, luego a la libertad y después a la eficacia: o sea, a convertirse en un nuevo sistema, dado que ahora la institución incorpora tecnología y se ha reorganizado.

Estudiando el mismo asunto -las instituciones sociales y su mal funcionamiento-, Illich, un crítico social, en su libro "La convivencialidad" llega a imágenes diferentes, que nada tienen que ver con la estabilidad, y sí en cambio con la opresión del ciudadano por parte de esas instituciones. La tecnología (más exactamente el exceso de tecnología), pero en particular la tecnología derivada de la cibernética, permitirá a los mismos Hs de Beer dominar ("esclavizar") mejor al personal. Es decir, en la visión de Illich H es la "nomenclatura" de la institución o el especialista, la tecnología cibernética es I, y O es el ciudadano de a pie. Como puede deducir fácilmente el lector, tanto Illich como Beer, además de partir de objetivos diferentes -en uno es la crítica social, en el otro la propuesta de soluciones activas- aplican en sus observaciones una ideología distinta. Podemos sostener, sin temor a errar, que en el caso concreto de Beer el enfoque de sistemas ha pasado de ser un instrumento I externo a convertirse en un factor ideológico personal.

Otro marco específico que podemos comentar un momento a la luz del U.F.M. es nuestro modelo de complejidad de tres niveles (véase capítulo de **Marcos conceptuales y Anexos**), que venimos aplicando fructíferamente a la informática y a la ofimática.

Todo el modelo es un **I** diseñado inicialmente para dar a los técnicos de la información de alta cualificación (el **H** implícito) una visión sociotécnica de un objeto constituido por la tecnología junto con su proceso de implantación en las instituciones sociales. Sus tres niveles se corresponden con tres esquemas típicos de complejidad que el modelo pretende abarcar en una visión integral. En el primero, el objeto es un elemento tecnológico (convencionalmente) aislado, como puede ser un ordenador o un programa, y su observador es un especialista, que utiliza sus técnicas propias. Ahí emerge una complejidad especializada, como por ejemplo la complejidad computacional.

El segundo nivel considera un objeto formado por elementos del anterior interrelacionados, situación que es la propia a manejar por un jefe de proyecto o un director de sistemas. Con el tercer nivel se configura una tripleta **H, I, O** que dibuja situaciones de complejidad plenamente sociotécnica, situadas bajo la responsabilidad de un director de informática o del director de la institución, que observa (diseña, construye, etc) con ciertas herramientas -tal vez con la ayuda de metodologías blandas como la Soft Systems Methodology de Checkland- un objeto formado por la interacción propositiva de un sistema tecnológico y de un sistema social.

Por su alto grado de abstracción y su sencillez conceptual, el modelo de los tres niveles es también un instrumento adecuado para directivos no profesionales de la tecnología, además de para tecnólogos, quienes, como se ha dicho, son sus destinatarios principales.

Muchos otros marcos se han creado a los que ahora podemos "observar" con nuestro esquema $H \times I \times O$. Pueden ser tanto instrumentos físicos como cognitivos. Citemos, por su actualidad, toda la tecnología CSCW (Computer Supported Cooperative Work), cuyos productos son a fin de cuentas herramientas para manejar la complejidad de interacciones en un grupo humano. De hecho, algún producto muy concreto es una interfaz que la técnica proporciona a los ejecutivos para que éstos "vean" el objeto ordenador como el potente soporte de una red de conversaciones.

Los entornos de interfaz **WYSIWYG** (**W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et) y en general toda la tecnología derivada del campo ahora conocido como **C.H.I.** (Computer Human Interaction), las técnicas relacionales para bases de datos, los sistemas expertos y otros muchos desarrollos, enfoques o teorías pueden considerarse útilmente, por lo menos en un plano conceptual, desde una perspectiva **U.F.M.** También sería útil como ejercicio que el lector relejera los capítulos anteriores intentando consolidarlos bajo esta perspectiva, aunque pueda resultar especialmente difícil en el caso de los textos, tan densos, del capítulo anterior.

A diferencia de los capítulos previos, en los que siguen, dedicados expresamente a algunos aspectos de las tecnologías de la información, haremos casi siempre explícita una breve interpretación dentro del marco U.F.M.

5. Resumen

Hemos expuesto un modelo de modelos -un marco para la construcción de modelos-, de aplicación universal, al que hemos denominado U.F.M.

Se expresa por la fórmula $H \times I \times O = IO$, cuyos términos hemos analizado aisladamente, interactivamente y recursivamente. Para tratar de hacerla comprensible y usable se han detallado algunos posibles componentes, valores o dimensiones de los términos de esta fórmula, y comentado diversos casos prácticos.

Esta fórmula es un programa epistemológico que puede guiar la acción de conocer la realidad (el objeto), pero también es un marco para la acción sobre ella, función ésta genuinamente específica de la ingeniería. Por tanto, es una guía para determinar las causas, condiciones y efectos de la complejidad, y, consiguientemente, sus tipos.

En este último sentido, hemos podido comprobar que la fórmula introduce muy directamente una clarificación en el tratamiento de la complejidad, al que hoy por hoy es imposible encauzar según una teoría unificada o algo parecido. Desafortunadamente, los autores que desarrollan teorías, métodos y técnicas o los diseñadores que usan algún procedimiento no suelen desvelar nítidamente ninguno de los términos de su esquema, con lo que siembran confusión, dejando en una nebulosa los límites de validez de su trabajo o de sus propuestas.

Por último, merece subrayarse la especial relevancia que para el estudio de la complejidad en su relación con la tecnología tiene para nosotros **-por su carácter de magnitud controlable-** el término **I**, tanto si se trata de herramientas físicas como de herramientas cognitivas.

Bloque III: Tecnologías de la información

Capítulo 11: Las tecnologías de la información

El conjunto de las tecnologías de la información constituye un gran laberinto de conceptos, técnicas, productos, dispositivos y sistemas. Por consiguiente, su consideración a vista de pájaro, para que no sea trivial, tiene que basarse en un modelo conceptual a la vez amplio y denso.

Aquí se presenta dicho modelo basado en unos procesadores abstractos y tres vectores de integración. Con este modelo inicial puede atacarse la construcción de un enfoque sociotécnico general de las tecnologías de la información y elaborar nuevos modelos, como el de los tres niveles de complejidad, ya visto anteriormente en este texto.

Todo ello configura una imagen, en nuestra opinión bastante potente e integrada, del espectro de las tecnologías.



1. Introducción

Hasta el momento nos hemos dedicado a sentar las bases teóricas necesarias para comprender la problemática relacionada con la complejidad. Se han analizado con cierta profundidad conceptos como los de sistema, variedad, jerarquía,... y, en general, el tema de la complejidad. Por lo tanto, el lector que haya llegado hasta esta página (y no se haya perdido por el camino) debe estar ya bastante familiarizado con todo el "mundo de lo complejo". Nos encontramos, por lo tanto, en condiciones de abordar el estudio, desde este punto de vista, de los casos prácticos y reales que anticipaba el título de este trabajo.

Pero para ello es preciso todavía introducir un nuevo elemento. En torno a él girará el resto de este trabajo: **las tecnologías de la información**. Antes de hablar de los problemas que plantean, y de los enfoques empleados para solucionarlos, será necesario un cierto estudio previo del término, de lo que engloba, del marco en el que se mueve, y de cómo se relaciona todo ello con el mundo de la complejidad. Es lo que haremos en este capítulo.

2. ¿Qué son las tecnologías de la información?

Lo mejor, naturalmente, será comenzar por el principio. Y el principio de nuestro estudio va a ser la expresión "tecnologías de la información"¹. Aunque es un término cada vez más utilizado, no lo es siempre con el mismo significado. Intentaremos aclarar el sentido con que se usará de aquí en adelante a partir de dos definiciones:

- a. [Sáez Vacas, 1983]: **"Tecnologías de la información son las que se aplican en la adquisición, procesamiento, almacenamiento y diseminación de información vocal, icónica, textual o numérica"**²
- b. [Valle, 1986]: **"Se consideran tecnologías de la información aquéllas cuyo propósito es el manejo y tratamiento de la información, entendida ésta como conjunto de datos, señales o conocimientos, registrados o transportados sobre soportes físicos de muy diversos tipos. Las tecnologías de la información abarcan técnicas, dispositivos y métodos que permiten obtener, transmitir, reproducir, transformar y combinar dichos datos, señales o conocimientos."**

El lector, tras una breve reflexión, estará seguramente de acuerdo en que ambas definiciones dibujan un significado muy similar. Y entre las dos, de una forma complementaria más que antagónica, cubren prácticamente el amplio espectro de las T.I. Nosotros, sin embargo, y para fines prácticos, de cara a saber a qué nos referimos en el resto de este trabajo, podemos quedarnos con una idea simplificada, que está contenida directamente en ellas. La idea de que las T.I. son las tecnologías que sirven para procesar, de alguna forma, algún tipo de información. Siendo mucho menos rigurosa que las expuestas en las dos definiciones, esta idea será, con toda seguridad, suficiente para nuestros propósitos. En los apartados siguientes, intentaremos profundizar en unos cuantos aspectos fundamentales que se derivan de esta idea (y también, directamente, de las dos definiciones dadas hace un momento).

¹Por abreviar, algunas veces nos referiremos a ellas simplemente como T. I.

²Continúa un poco más adelante F. Sáez Vacas: "Esta definición es incompleta puesto que deja fuera del discurso un tipo fundamental de información, que propongo incluir, la información sobre el mundo de la materia".

3. Los procesadores de información

Vamos a tratar de construir una clasificación de los procesadores de información (entendidos como máquinas capaces de manipularla de alguna forma). En general, podemos decir³ que cualquier manipulación de la información como si fuera un objeto abstracto consistirá en una combinación de tres básicas: traslación en el tiempo (almacenamiento), traslación en el espacio (transporte) o traslación en la forma (cambio de morfología). Llamaremos **procesador T**, **procesador E** y **procesador F** a los procesadores ideales que solo realizan traslación en una de las dimensiones señaladas (respectivamente, tiempo, espacio y forma).

Desde este marco, podemos clasificar fácilmente a cualquier aparato o tecnología que sirva para tratar información. Por ejemplo, el teléfono es principalmente un procesador *E*, que actúa (de cara a los usuarios) sobre la información sin almacenarla ni modificarla (idealmente), sólo transportándola del micrófono al hilo telefónico. Un circuito codificador es un procesador *F*. Y los procesadores *T* por excelencia son las memorias, que conservan la información "congelada" en el espacio y en la forma, desplazándola sólo en el tiempo.

Lo mismo podemos hacer para sistemas más complejos. Aunque generalmente estén compuestos por una combinación de elementos *T*, *E* y *F*, muchas veces su funcionalidad básica, la que percibe el usuario⁴, permite verlos como un procesador simple. Así, la red telefónica se comporta idealmente como un procesador *E* (toma información y la transporta en el espacio), aunque esté compuesto por un gran número de procesadores de todos los tipos⁵.

Hay un sistema de información que no puede caracterizarse mediante un solo tipo de procesador. Es el ordenador. Su funcionamiento consiste a la vez en traslaciones en el tiempo (memoria) y en la forma (proceso de información) controladas por el programa. Es por tanto un ejemplo claro de procesador *TF*.

³Como nos indica F. Sáez Vacas. Todo este apartado está tomado de su artículo [Sáez Vacas, 1983].

⁴Podríamos decir que adoptamos aquí una visión funcional del sistema. Aunque "por dentro" haya muchos componentes, realizamos abstracción de ellos, y nos fijamos sólo en su "función de transferencias".

⁵En realidad, el usuario de la red telefónica percibe muchas veces también indeseables efectos debidos a que no es en realidad un procesador *E* puro, sino que también tiene componentes *T* (manifiesta en los retardos) y *F* (distorsiones).

Pero no para aquí el asunto. Cada vez más, se tiende a la interconexión de ordenadores mediante redes de transmisión de datos. Unimos a la capacidad de almacenamiento y proceso de la información que provee el ordenador, la traslación en el espacio. Surgen las "redes telemáticas", máximo exponente de lo que suponen las tecnologías de la información, proveen al usuario con un verdadero **procesador TEF**. Se unen así en las T.I. todas las posibilidades de manipulación de información, proporcionando al usuario humano unas herramientas extraordinariamente potentes. Pero, a la vez, y precisamente por su gran potencia y versatilidad, nada fáciles de manejar.

4. Vectores de integración

Hemos hablado de la funcionalidad de las tecnologías de la información. Hablemos ahora de su estructura. Y para analizarla, para entender el gran desarrollo de los últimos años y la gran problemática que plantean las T.I. en la actualidad, exponeremos el modelo de los "**tres vectores de integración**"⁶. La idea central es que el gran impacto que están teniendo estas tecnologías ha sido posible por la confluencia de tres vectores:

- a. El **vector de electrificación**.
- b. El **vector de digitalización**.
- c. El **vector de computadorización**.

Desde luego, estos vectores no sólo confluyen "independientemente", sino que las interrelaciones a las que están sujetos son innumerables. Detengámonos brevemente en cada uno de ellos.

El papel de la electrónica es, hoy por hoy, indudable en el desarrollo de las T.I. Los rápidos avances que ha experimentado ésta han hecho posible la aparición, a un coste económico cada vez menor, de dispositivos más y más potentes con un menor consumo de energía. Al ser baratos, su diseminación y penetración en casi todos los mercados ha sido rápida. Al ser más potentes, permiten cada vez mejores prestaciones de los aparatos en los que se usan, medidas éstas por su velocidad de proceso, capacidad de memoria o de transmisión, versatilidad, etc.

⁶Original de F. Sáez Vacas, expuesto ya en [Sáez Vacas, 1983] y completado posteriormente por el autor en explicaciones de clase, artículos y libros.

La digitalización proporciona una cierta uniformización de la información, en el sentido de que, de una forma o de otra, al final todo se va a reducir a una serie de bits (naturalmente, para el caso de la digitalización binaria). Da lo mismo que estemos tratando con imagen, con sonido, con voz o con cotizaciones de bolsa. Podemos por lo tanto aplicar una bases teóricas muy similares, bastante independientes del significado de la señal que estamos manejando. Y además, éste es un soporte especialmente adecuado para tratarlo con medios electrónicos. El efecto de sinergia entre estos dos vectores es aquí fortísimo.

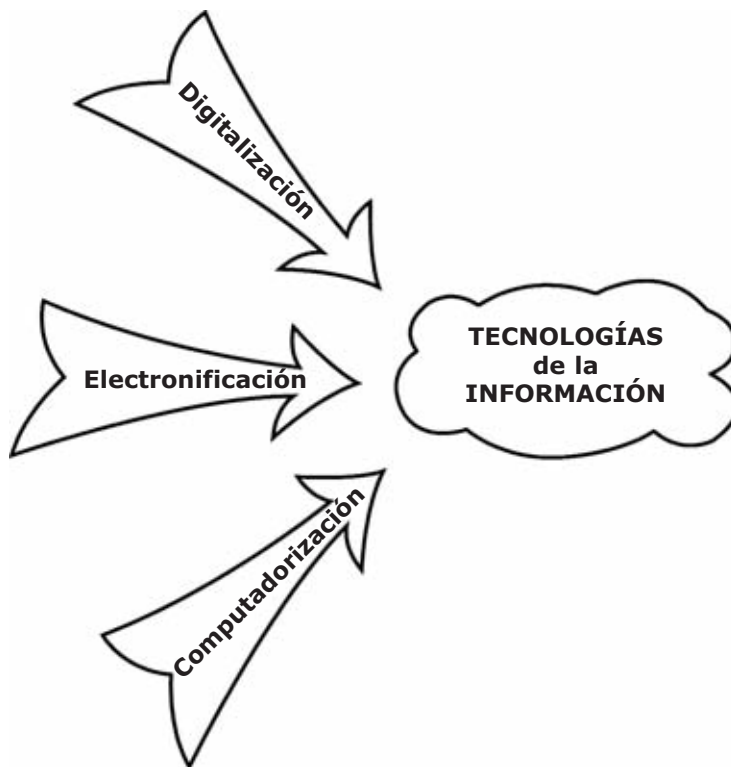


Fig. 1. Los tres vectores de integración de las tecnologías de la información (F. Sáez Vacas, 1983)

Y nos falta el último (y el más importante) de nuestros factores. La aparición del ordenador, su desarrollo y su "democratización" de la mano del microprocesador, han hecho posible un efecto de bola de nieve en el que hoy día nos hallamos totalmente inmersos. Producto, en cierta forma, de los dos vectores anteriores (ya que el ordenador es a la vez electrónico y digital), ha servido para relanzarlos hasta cotas insospechadas. Por ejemplo, el rápido desarrollo de la microelectrónica no sería posible sin la ayuda de potentes sistemas informáticos de ayuda al diseño⁷. Y la aplicación de ciertos algoritmos de tratamiento de la señal (que permiten manejar eficientemente información digital) es sencillamente imposible sin el uso de microprocesadores.

El que las T.I. sólo dependan básicamente de la confluencia de estos tres factores facilita la integración y la interconexión de elementos apoyados en ellas. Como todos tienen soportes materiales similares (electrónica, señales digitales, ordenadores), pueden combinarse con facilidad. Desde el modelo TEF, esto permite un espectacular crecimiento de las posibilidades de combinación entre procesadores del tipo que sea para crear otros más complejos (y cada vez más "inteligentes"). Así crece también el conjunto de funciones generadas por la tecnología, al tiempo que los métodos que utilizan tienden a la normalización⁸.

5. La complejidad del mundo y la complejidad de las T.I

Toda esta evolución nos lleva, sin duda, hacia unos medios tecnológicos más potentes, más eficaces, más versátiles, pero también más complicados, más difíciles de entender en sus muchas interrelaciones, más difíciles de controlar. Y aún hemos de introducir en el sistema tecnológico al ser humano, como usuario final de las T.I., y analizar sus relaciones con ellas, los problemas de la interacción hombre-máquina, las consecuencias sociales de esta interacción... Todo se va haciendo cada vez más complejo.

⁷"... es inimaginable pensar los resultados actuales [en microelectrónica] sin el apoyo de unas máquinas, ordenadores, etc., cuya existencia es fruto de desarrollos microelectrónicos anteriores que, a su vez, se apoyaban en otras máquinas propiciadas por otros resultados previos, y así sucesivamente"(tomado de [Valle, 1986], págs. 92-93).

⁸El efecto de la confluencia de los tres vectores mencionados y de la combinación de procesadores TEF puede apreciarse en todos los elementos que componen el ancho mundo de las tecnologías de la información. Las redes telefónicas se van llenando de ordenadores que hacen la función de central telefónica, a la vez que la voz se digitaliza cada vez en más enlaces. Aparecen y se desarrollan las redes de conmutación de paquetes, fruto de una estrecha colaboración entre electrónica, informática y ordenadores. Los aparatos telefónicos avanzan hacia una configuración como terminales genéricos que permitan el almacenamiento y el proceso de datos de todo tipo...

Pero a la vez hemos de considerar que nos movemos también en un mundo complejo, como ya hemos ido viendo en los capítulos precedentes. Si introducimos este nuevo factor, podría quedarnos un esquema conceptual como el de la figura 2, un bosquejo de distintas relaciones **entre la complejidad del mundo, la complejidad de las tecnologías de la información y la aplicación de estas tecnologías**⁹. A continuación, exponemos brevemente las principales características de estas relaciones:

- a. **Impacto de la complejidad de las T.I. sobre la complejidad del mundo.**
Está claro que la aplicación de nuevas tecnologías tiene un impacto notable sobre la sociedad. De una forma general, producen nuevas interrelaciones, profundizan el conocimiento, crean desequilibrios, favorecen ciertos procesos, de forma que aumentan la complejidad del mundo con que tratamos.
- b. **Necesidad del estudio de la complejidad del mundo antes de pasar a la aplicación de la T.I.** Naturalmente, la complejidad de la sociedad, del mundo que nos rodea, de las instituciones que en él nos encontramos condicionan la aplicación que se pueda hacer de las T.I. Es preciso conocer bien las características de esta complejidad para que los efectos conseguidos al aplicar T.I. sean los deseados.
- c. **Necesidad del estudio de la complejidad de las T.I. antes de aplicarlas.** Esta relación hace referencia directa a la complejidad de las T.I. "por sí mismas", que, como vamos viendo, es muy grande. Es preciso conocerla en profundidad para ser capaces de aplicarlas con efectividad.

⁹Tanto el esquema de la figura 2 como la discusión que de él se hace están tomados del artículo [Sáez Vacas, 1987].

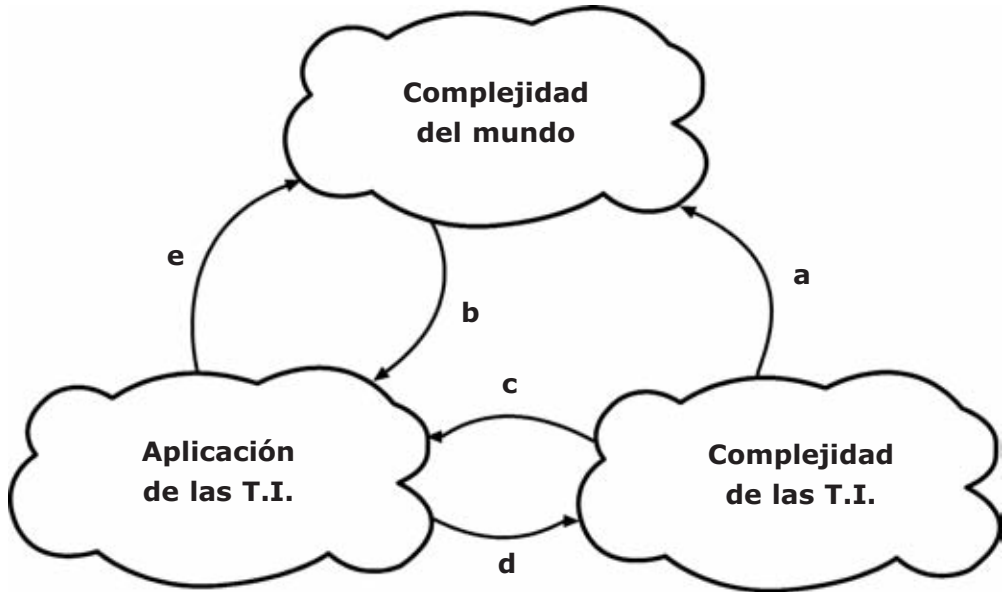


Figura. 2. Relaciones de complejidad al introducir las T.I. en un marco humano [Sáez Vacas, 1987]

- d. **Influencia de la aplicación de las T.I. en su misma complejidad.** La aplicación de las T.I. puede complicar considerablemente la propia dinámica de las T.I. Será necesario que, a la hora de diseñar las propias tecnologías, tengamos una idea clara de cómo van a ser usadas, para intentar que esta influencia nos lleve por el contrario a una cierta simplificación. Por ejemplo, el desarrollo de mejores interfaces hombre-máquina hará sin duda que la complejidad que perciba un usuario de las T.I. sea mucho menor.
- e. **Papel de la aplicación de las T.I. en el manejo de la complejidad del mundo.** Las tecnologías de la información pueden ayudarnos a manejar la creciente complejidad del mundo. Pero para ello hemos de utilizarlas adecuadamente, pues también pueden producir el efecto contrario. Pensemos en la introducción de un sistema informático como ayuda a un equipo de toma de decisiones¹⁰. El sistema puede ser diseñado de forma que tome una gran cantidad de datos de la realidad sobre la que hay que decidir, procese esta información, y dé soporte al equipo exponiendo los pros y los contras de las distintas

¹⁰Este ejemplo está inspirado en uno similar propuesto en [Beer,1974].

decisiones a tomar (mediante simulaciones, análisis estadísticos, etc.). De esta forma funcionaría como filtro de la variedad, presentando a los seres humanos algo de menos complejidad, con la menor pérdida de información posible. Pero, ¡ojo!, también podría ocurrir que el sistema estuviese mal diseñado y proporcionase únicamente inmensos listados (eso sí, muy actualizados y detallados), que el grupo simplemente no podría abarcar. En ese caso, aumentaría aún más la complejidad del mundo que perciben los encargados de la toma de decisiones, hasta tal punto que no pudieran absorber tanta variedad. Desgraciadamente, esta situación no es puramente ficticia.

Estas mismas relaciones pueden contemplarse a la luz del **modelo de los tres niveles de complejidad** propuesto por F. Sáez Vacas. Mientras que los sistemas que utilizan T.I., tomados aisladamente, pueden incluirse en el segundo nivel (complejidad sistémica), al introducirlos en la sociedad pasamos a estar ante un problema englobable en el tercer nivel (complejidad antropotécnica). Y en este nivel, como ya hemos estudiado, emergen nuevas propiedades y relaciones, con la problemática de la relación hombre-máquina en lugar destacado. Las relaciones que se han visto esquematizadas en la figura 2 ejemplifican muy bien la nueva problemática.

6. Las tecnologías de la información y el modelo $H \times I \times O = IO$.

Podemos enfocar el estudio de las T.I. que hemos realizado en estas páginas desde el marco $H \times I \times O \rightarrow IO$ de Sáez Vacas. Analicemos cada uno de los elementos que lo componen, para el caso particular de las T.I.:

- a. El **objeto**. Como tal consideraremos, de una forma global, a las tecnologías de la información en su conjunto.
- b. El **observador**. En principio, puede ser cualquier persona interesada en obtener una visión de las tendencias de las T.I., y sus relaciones con la sociedad. Sólo serán requisitos necesarios el tener una cierta preparación que permita entender al menos los rudimentos de los problemas de complejidad¹¹, y algunos conocimientos puramente técnicos que son precisos en el campo de las T.I.
- c. La **interfaz**. Se han utilizado simultáneamente dos modelos conceptuales: el de los tres vectores de integración (producto de una observación histórico-empírica) y el de los procesadores abstractos TEF.

¹¹El haber entendido lo tratado en los primeros capítulos de este trabajo asegura al lector, en cierta medida, que dispone de esta preparación.

- d. La **imagen del objeto**. Como resultado del proceso de modelación, obtenemos una visión de las T.I. donde destacan dos aspectos, fuertemente interrelacionados. Por un lado, destaca la presencia de un entrelazamiento de procesadores cada vez más grande y complejo. Las posibilidades de combinación de elementos TEF produce esta "explosión". Por otro, un conjunto cada vez más reducido y potente de técnicas de diseño, construcción y gestión de las T.I. (como fruto directo de la integración de los tres vectores ya discutidos).

La imagen que hemos obtenido, trasladada al entorno social donde se insertan las T.I. (según la implicaciones que se han deducido del esquema de la figura 2) nos lleva a otras necesidades. Entre ellas, cabe destacar las técnicas cibernéticas y las que proporciona la visión sistémica del entorno. También aparecerán nuevos modelos consistentes con este nuevo planteamiento donde el objeto de interés son las T.I. en la sociedad. Entre ellos cabe destacar el ya expuesto modelo de tres niveles de Sáez Vacas.

7. Resumen

Por fin hemos llegado al objetivo central de nuestro estudio: las tecnologías de la información. En los próximos capítulos estudiaremos en detalle casos concretos de aplicación de estudios de complejidad sobre diferentes parcelas. Pero para comenzar, y hacerlo de una forma rigurosa, nada mejor que unas **definiciones** que centren el campo de estudio.

Después de proponerlas, y analizarlas brevemente, se expone el **modelo de los procesadores TEF**, que proporciona un marco adecuado para comprender las características principales de las T.I., y sus consecuencias. Y además, nos va dejando entrever ya el alto grado de complejidad con el que en ellas nos encontraremos.

Por último, se ha puesto de manifiesto la problemática surgida de la **integración de las T.I. con el entramado social**. No podemos verlas como sistemas aislados, sino que nos vemos obligados a considerar también las interacciones con los seres humanos que, al final, van a disfrutar (o padecer) sus efectos. De aquí emerge un nuevo "tipo" de complejidad, precisamente la contemplada en el tercero de los tres niveles del modelo propuesto por Sáez Vacas.

8. Bibliografía

Dividida en dos partes. En primer lugar, Notas Bibliográficas, donde se describen los trabajos consultados más relevantes sobre el tema. Después, Referencias Bibliográficas, donde pueden encontrarse todas las citas utilizadas en el capítulo.

Notas bibliográficas

El artículo [Sáez Vacas, 1983] está dirigido a un público técnico no especializado. Tiene carácter divulgativo, aunque conceptualmente es muy denso. Constituye la base inicial de este capítulo y en él puede encontrarse muy útilmente un razonamiento completo de lo aquí expuesto.

[Valle, 1986] es un texto que hace un repaso, desde un punto de vista similar al de Sáez Vacas (aunque quizás menos estructurado) al mundo de las tecnologías de la información. Es especialmente aconsejable para el lector que desee tener una amplia panorámica del mismo.

En [Sáez Vacas, 1987] se estudian las relaciones entre las tecnologías de la información, la complejidad y el mundo. De este artículo están tomadas las ideas que se han expuesto sobre este particular en el presente capítulo.

Referencias Bibliográficas

Beer, S. (1974): **Designing Freedom**, Wiley & Sons, Londres.

Sáez Vacas, F. (1983): "Las tecnologías de la tercera revolución de la información", **Mundo electrónico**, núm.183, pág.133-141.

Sáez Vacas, F. (1987): "Towards a conceptual remodeling of information technologies based on a broad consideration of complexity", comunicación en el **31st Annual Meeting of the International Society for General Systems Research**, Budapest, 1-5 de junio de 1987.

Sáez Vacas, F. (1990): "A complexity architecture for information technologies: a three years didactic experiment", **Systems Practice**, vol.3, núm.1, pág.81-96.

Valle, R., Ros, F., Barberá, J. y Gamella, M. (departamento de promoción tecnológica de FUNDESCO) (1986): "Tecnologías de la información: electrónica, informática y telecomunicaciones", editado en **Notas del curso "Fundamentos y función de la ingeniería"**, ETSI Telecomunicación, Madrid (tomado del libro **Los países industrializados ante las nuevas tecnologías**, FUNDESCO).

Capítulo 12: Hardware

*La tecnología microelectrónica ha dado un impulso tan espectacular a la estructura y arquitectura de los ordenadores que ha cambiado en pocos años varias veces los órdenes de magnitud de sus prestaciones y su complejidad. Cada uno de los temas que se tocan en este capítulo, y otros que ni siquiera incluimos, constituyen de por sí un dominio especializado. Por esta razón, nos hemos limitado a abordarlos de forma que se resalten los aspectos que tienen que ver con conceptos como **integración, niveles jerárquicos, capas, clasificaciones, categorías de máquinas, relaciones hardware-software y paralelismo.***



1. Introducción

En los capítulos anteriores hemos ido desarrollando un marco conceptual sobre complejidad y sistemas que vamos a tomar ahora como punto de partida para repasar las tecnologías hardware. Normalmente, cuando se habla de hardware, nos referimos a los ordenadores en su aspecto físico, cómo están contruidos, características de diseño, materiales empleados, organización interna, etc.. Aunque a la hora de trabajar con ellos estas características queden ocultas por diversas capas de máquinas virtuales y el software de aplicación, es fundamental para el ingeniero poseer unos conocimientos básicos sobre arquitectura y aspectos físicos del computador. Esto no es un gran descubrimiento pero sí requiere un análisis detallado para evitar aproximaciones simplistas, pues el hardware se ha convertido en una tecnología muy compleja que a fuerza de avanzar a pasos agigantados se ha diversificado de tal manera que es muy difícil obtener una perspectiva global. Hace muy pocos años los ordenadores diferían en muy pocas cosas y casi se podían contar con los dedos de la mano las diferentes aproximaciones prácticas al problema de la computación. Sin embargo, hoy en día existen ordenadores comerciales que funcionan basándose en interpretaciones radicalmente distintas y la creciente tendencia a integrar software y hardware acentúa aún más la diversificación.

Si se intenta dibujar una perspectiva generalista, no hay más remedio que recurrir a las herramientas que hemos ido detallando en capítulos anteriores. No en vano, muchos de los trabajos sobre complejidad y sistemas comentados proceden de investigadores y científicos muy relacionados con los ordenadores. Se puede cerrar un bucle imaginario que va desde el estudio de la complejidad a los ordenadores y de éstos a la complejidad, pues, como se ha dicho, "el ordenador es el instrumento de las ciencias de la complejidad" [Pagels, 1989, p. 36]. En el

presente capítulo intentaremos trazar un enfoque sistémico y generalista de las tecnologías de hardware haciendo hincapié en la relación que tienen muchos de los aspectos que trataremos con las nociones de complejidad y sistemas. La referencia a ideas ya mencionadas será obligada e invitamos al lector a encontrar otras nuevas y a interpretar las tecnologías que presentamos a la luz del estudio de la complejidad.

Es una opinión extendida dentro del mundillo informático que las auténticas "revoluciones" tecnológicas siempre vienen y han venido de la mano de avances en el hardware. El resto de la tecnología, incluido el software, se desarrolla y evoluciona según la pauta que marcan los equipos y plantean la innovación en función de éstos. Existen muchos ejemplos que demuestran lo acertado de esta opinión, uno de ellos es el creciente interés en la publicación electrónica dada la aparición de muchos programas que aprovechan los nuevos monitores gráficos (VGA en los pc's, por ejemplo) y las prestaciones de las impresoras láser, ahora con un precio suficientemente bajo como para pensar en conectarlas a ordenadores personales.

Pero un prueba aún más palpable es que el software va siempre por detrás del hardware, en investigación, desarrollo y productos comerciales. Muchos equipos ven hipotecadas sus prestaciones por la falta de programas adecuados, uno de los pilares básicos de la industria de los pc's es la cantidad de programas disponibles, un hecho que actúa como serio obstáculo para introducir nuevos modelos de ordenador. En ordenadores de la gama alta (superordenadores y máquinas especializadas) el problema es similar y gran parte de la investigación se ve frenada por la falta de lenguajes de programación adecuados y herramientas que puedan sacar partido a las nuevas arquitecturas. Este fenómeno se puede comprobar en los ordenadores paralelos donde existe ya una cantidad apreciable de hardware comercial y, sin embargo, el estado del arte del software paralelo no permite utilizar adecuadamente muchas de ellas.

Y como demuestra el ejemplo que mencionábamos ahora sobre la publicación electrónica, las aportaciones del hardware no se producen sólo en el terreno de la arquitectura de ordenadores, sino también en el de coprocesadores, impresoras, monitores gráficos, redes de comunicación, medios de almacenamiento y todo tipo de periféricos. Gran parte de las ideas sobre hipertexto, por ejemplo, serían impensables si no fuera por el aumento de capacidad de los discos duros y la aparición de los discos ópticos, muchas aplicaciones de diseño asistido no se hubieran desarrollado de no existir los periféricos adecuados, desde monitores de alta resolución hasta *plotters* y tableros gráficos, etc.

2. Hardware-Software

El estudio de los ordenadores se ha dividido tradicionalmente en hardware y software. La primera disciplina se centra en los recursos físicos para la computación y va desde el estudio de los dispositivos electrónicos hasta las arquitecturas de los ordenadores. El software se suele identificar con la programación de esos recursos físicos para conseguir que realicen determinadas tareas. Sin embargo, la frontera no es muy clara en bastantes ocasiones, sobre todo dada la tendencia a integrar software y hardware, un punto sobre el que volveremos más adelante.

2.1 Máquinas virtuales o el hardware fantasma

La separación existente entre el hardware y el software plantea una dificultad evidente en el manejo de los ordenadores. Esta es una de las primeras ideas que se estudia en "fundamentos de ordenadores". En el hardware se manejan, dependiendo del nivel, desde coeficientes de amplificación y capacidades de sustrato a ciclos de bus, velocidades de acceso y capacidad de memoria. En el software, el nivel inferior maneja direcciones de memoria, estado del procesador y "flags" de condiciones, componentes que definen un nivel muy por encima de lo que es el hardware. Existe un nivel intermedio, el de microprograma, que es difícil de encajar tanto en el hardware como en el software y en el que se produce la simbiosis entre ambas vertientes de la computación. A partir de él se empieza a desarrollar el software en sucesivos niveles, según la aproximación clásica. Cada uno de esos niveles se interpreta desde el inmediato superior como una máquina, como si fuera hardware, de ahí el nombre que reciben de máquinas virtuales.

Esta estructura "de cebolla" es la que aparece en la siguiente figura con la distribución en capas de un sistema operativo:



Fig.1. Sistema operativo construido a base de capas, o máquinas virtuales.
[Fernández, Sáez Vacas, 1984, p. 717]

Esta misma estructura jerárquica en capas la vamos a encontrar en el hardware, tal y como recogíamos en un capítulo anterior (ver "Sistemas, visión estructural y visión funcional", en particular la figura 7).

Hardware y niveles de complejidad

En el capítulo dedicado a los marcos conceptuales recogíamos un modelo de complejidad propuesto por Sáez Vacas [1983] que planteaba tres niveles en la informática. Un primer nivel es el de los objetos o elementos aislados, como puede ser un algoritmo, un circuito electrónico, etc. Estos elementos así considerados exhiben una complejidad característica que se denomina complejidad de los elementos aislados. Por encima de este nivel está la complejidad de los sistemas, es decir, la complejidad que aparece cuando esos elementos aislados se combinan para formar un todo del que interesa su comportamiento global. Un ejemplo es un ordenador personal que puede verse como un sistema compuesto de un gran número de elementos. Otro puede ser un chip. El último nivel de este modelo introduce un tipo de complejidad diferente a la que aparece en los otros dos, es la complejidad antropotécnica que aparece como resultado de la interacción entre los sistemas tecnológicos y la sociedad. En este nivel es donde se estudian problemas tan importantes como las interfaces de usuario, los factores humanos en la implementación tecnológica, los cambios organizativos como consecuencia de la utilización de la tecnología, etc.

Cada una de las capas tiene un lenguaje propio y define un dominio de trabajo único para ese nivel. Esta jerarquía es importante porque da una idea muy clara de cómo se orquestan los diferentes niveles y las distintas tecnologías dentro de un mismo campo, la informática, y puede ser muy útil para interpretar los avances y desarrollos que se producen. Por ejemplo, se puede pensar en integrar varias capas en circuitos electrónicos (como de hecho está sucediendo cada vez más), o trasladar la complejidad de una capa a otra, simularlas en un ordenador distinto, etc.

La razón para utilizar este tipo de jerarquías ya la vimos en el capítulo dedicado al tratamiento de la complejidad, pero conviene recordar que estas clasificaciones nos proporcionan un mapa muy útil del terreno por el que nos movemos y nos permiten establecer relaciones y niveles bien diferenciados para poder estudiarlos por separado.

2.2 Integración de Hardware y Software

Dada la semejanza en la estructura adoptada para interpretar el hardware y el software, se puede pensar en cambiar los límites establecidos entre ambos campos. Se podría, por ejemplo, implementar en hardware alguna de las capas que forman el sistema operativo, o incluso ir más lejos e implementar directamente el sistema operativo en forma de microprograma. En la realidad se ha avanzado aún más, existen ordenadores cuya arquitectura está directamente orientada a un lenguaje de programación concreto y las instrucciones se ejecutan sobre la máquina directamente.

De la misma forma, se puede pensar en reproducir mediante programas diferentes niveles de hardware, como de hecho se hace en las máquinas que simulan un ordenador distinto. La estructura en capas permite pensar en un proceso de computación en el que la frontera entre el hardware y el software se deja a elección del diseñador, de ahí la importancia de tener una perspectiva de conjunto.

Más adelante volveremos sobre las arquitecturas dedicadas (construidas para un lenguaje o una aplicación concreta), por ahora mencionemos sólo que cada vez son más los ordenadores comerciales que abandonan las arquitecturas tradicionales y optan por una estructura interna especialmente adaptada para el uso que se va a hacer de ellos.

Como ejemplo de la simulación mediante software de niveles hardware podemos citar las máquinas UNIX que ejecutan el sistema operativo DOS como una subtarea, lo que permite trabajar con todas las aplicaciones diseñadas para este sistema operativo en máquinas en las que, en principio, no se podrían utilizar. Como ejercicio piense el lector cómo y qué funciones deberían simularse para que el funcionamiento fuera idéntico al de una máquina DOS (como pistas mencionaremos la entrada/salida, dispositivos de interacción, y la velocidad de ejecución).

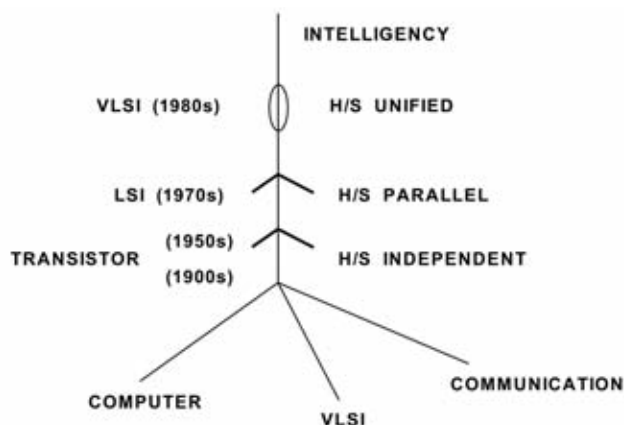


Fig. 2. Integración del Software y el Hardware [Matsumura, 1983]

Matsumura recoge esta misma idea de integración entre el hardware y el software y la resume en la figura 2.

Antes de acabar este apartado, conviene mencionar un hecho que, aunque no está directamente relacionado con la integración hardware y software, es muy importante para entender la evolución de los ordenadores. Gran parte de las investigaciones sobre nuevos materiales y dispositivos electrónicos dependen de la capacidad de computación disponible, es decir, cuando se dispone de tecnología para construir un ordenador más rápido, más potente y con más capacidad que los anteriores, se puede utilizar para perfeccionar las técnicas que permitieron su construcción, al mejorar éstas se puede desarrollar otro modelo más potente y así sucesivamente. Este bucle es responsable, al menos en parte, de los grandes avances que se han dado en las tecnologías de la información, se puede decir que se trata de construir una herramienta que nos permita construir una herramienta mejor. Ejemplo palpable de ello es la tecnología VLSI que permite diseñar ordenadores mucho más potentes y que, paradójicamente, es el resultado de poder utilizar computadores para gestionar la complejidad de los diseños de circuitos integrados.

Para muchos, el motor de los avances en las tecnologías de la información es el hardware.

2.3 VLSI

Hablar sobre hardware puede convertirse en una tarea muy ardua si no delimitamos el alcance de la discusión previamente. Aquí nos limitaremos a las arquitecturas de ordenadores, pero antes de entrar a estudiarlas, es conveniente detenerse un momento en los niveles más bajos del hardware, concretamente en el de circuitos.

Existe un consenso casi absoluto a la hora de atribuir el enorme avance de los ordenadores a la creciente capacidad de integrar circuitos. Que no consiste sólo en hacerlos cada vez más pequeños, sino también más rápidos, eficientes, baratos y fiables. Sin esta capacidad de integración, el auge de los pc's hubiera sido impensable (por precio, tamaño y prestaciones) pero tampoco hubieran podido plantearse los ordenadores paralelos (utilizar varios procesadores, a veces en cantidades masivas), el hardware tolerante a fallos (que funciona fundamentalmente por duplicación), los procesadores dedicados (para comunicaciones, para servidores de red), el proceso de señal (con arquitecturas dedicadas gracias a las cuales son posibles muchas aplicaciones de comunicaciones y tratamiento de imágenes), etc.

El poder integrar cada vez más circuitos en un espacio más pequeño no es un capricho. En primer lugar, la mayor rapidez permite construir ordenadores más eficientes y con unas prestaciones elevadas. El poder incluir muchas funciones en un solo chip se traduce en mayor fiabilidad en el proceso de información (un fallo dentro del chip es menos probable si se divide en varias partes y se comunican a

través de un bus) y, lo que es más importante desde nuestro punto de vista, reduce la complejidad a la que se enfrentan los niveles superiores al permitir que parte de esa complejidad se traslade a los circuitos.

Los primeros ordenadores eran máquinas enormes que ocupaban salas enteras y exigían un control absoluto de las condiciones de trabajo (temperatura, humedad, vibraciones, etc.), de ahí hasta los ordenadores portátiles que conocemos hoy en día se ha recorrido un largo camino. En aquellos ordenadores primitivos no existían muchos de los niveles que hoy consideramos "casi" imprescindibles, el software era prácticamente inexistente tal y como lo entendemos hoy en día, y el hardware era muy simple comparado incluso con el del un ordenador personal actual.

Escalas de integración

ULSI es, por ahora, el último escalón de una serie de tecnologías de integración de circuitos. Anteriormente, los ordenadores se basaban en tecnología SSI (pequeña escala de integración) o MSI (media escala de integración), con las que se llegaba a integrar hasta 10 puertas lógicas. Posteriormente se alcanzó la LSI (gran escala de integración) con la que se pueden integrar hasta 10.000 puertas lógicas y VLSI que permite integrar cientos de miles de puertas en un chip. ULSI integra millones de puertas.

El sucesivo desarrollo de los niveles de la jerarquía se hizo en función de la capacidad para integrar circuitos electrónicos en un espacio cada vez menor. Al mismo tiempo, había que reducir el consumo de potencia, la disipación de calor, flexibilizar las condiciones de trabajo y aumentar la velocidad de computación. Gran parte de los avances que se han producido en tecnologías de la información se deben a avances previos en el nivel más bajo del hardware, el de circuitos.

Hoy en día prácticamente existen ordenadores en un solo chip y la integración de la que hablábamos antes no sería posible sin esta tecnología. Gracias a ella se puede pensar en implementar en hardware cualquiera de los niveles de la jerarquía propuesta por Bell y Newell, incluso los niveles superiores, de aplicaciones, algo impensable hace tan sólo unos años. Gracias a ello, el diseñador de ordenadores ha de enfrentarse con una complejidad distinta ya que cuenta con procesadores y circuitos periféricos de gran potencia que resuelven gran parte de los problemas. Por ejemplo, gran parte de los procesadores avanzados de hoy en día llevan incorporada la Unidad Aritmético-Lógica a la CPU, ya no hay que diseñar la conexión entre ellos, lo mismo sucede con las memorias cache y los gestores y muchos otros elementos.

3. Clasificación de las arquitecturas de ordenadores

Dada la enorme velocidad de desarrollo de los ordenadores es difícil tener una imagen de conjunto de las cosas que existen, las que pueden existir, qué se investiga y qué aproximaciones han quedado superadas. El bombardeo de información, comercial e investigador, al que está sometido cualquiera que trabaje en este campo es suficiente para despistar al más experto. Ordenadores paralelos, procesadores en array, conexionismo, monoprocesadores, sistemas distribuidos, procesadores dedicados, ordenadores vectoriales, micro y macroparalelismo, máquinas de flujo de datos, quinta generación, ordenadores LISP, RISC y CISC, etc., son términos que aparecen frecuentemente no sólo en las publicaciones especializadas sino en cualquier folleto publicitario, los entienda el futuro usuario o no.

Semejante diversidad puede parecer sorprendente si se tiene en cuenta que todas esas arquitecturas se derivan de unos pocos modelos establecidos en los años 40. Pero la variedad es tal que se necesita una clasificación para poder moverse con un mínimo de seguridad en este terreno [Dasgupta, 1990, p. 65].

Por otro lado, las clasificaciones son muy útiles para poder establecer posibles líneas de investigación y evaluar el estado del arte, cada vez más complejo. Existen muchas clasificaciones de las arquitecturas de ordenadores pues los parámetros a seguir son muchísimos, y se necesitan perspectivas muy amplias para que la clasificación no se quede anticuada en muy poco tiempo. Esta característica, la capacidad de predicción, es fundamental en toda taxonomía. Pero la velocidad de desarrollo de las tecnologías de la información en conjunto hace realmente difícil predecir el futuro, por eso muchas de las clasificaciones se hacen intentando extender otras ya existentes y más antiguas, a las que el tiempo ha restado interés y resolución.

3.1 Datos vs. Control, una clasificación tradicional

Una de las clasificaciones más utilizada es la basada en la relación entre datos y control dentro del procesador. Esta clasificación se debe M.J. Flynn y se puede encontrar en cualquier libro sobre ordenadores. A pesar de su simplicidad, es casi la única de uso general.

El criterio de clasificación es el número de instrucciones que se procesan simultáneamente y sobre cuántos datos. De acuerdo con este criterio, se distinguen cuatro grandes grupos de arquitecturas:

- SISD (Single Instruction, Single Data), Una Instrucción, Un Dato. Las instrucciones se ejecutan una a una y procesan un único dato cada vez. Ésta es la arquitectura clásica (Von Neumann).

- SIMD (Single Instruction, Multiple Data) Una Instrucción, Datos Múltiples. Cada instrucción opera con varios datos al ejecutarse. Esta es la arquitectura de los procesadores en array y de muchas máquinas orientadas al proceso numérico (como las de procesamiento de señales o de imágenes), donde las operaciones a realizar con los datos son relativamente pocas pero hay que efectuarlas sobre muchos datos.
- MISD (Multiple Instructions, Single Data) Múltiples Instrucciones, un Único Dato. Supuestamente se ejecutan muchas instrucciones que operan con un único dato. Esta arquitectura no tiene sentido fuera del puramente formal.
- MIMD (Multiple Instructions, Multiple Data) Múltiples Instrucciones, Datos Múltiples. Ejecución simultánea de varias instrucciones que operan con varios datos. Esta es la arquitectura de los ordenadores paralelos.

Como es fácil ver, la clasificación es muy simple y permite pocas distinciones entre los ordenadores. Cuando se formuló (1972), todavía no existían ordenadores paralelos (al menos no a gran escala y con paralelismo masivo) y muy pocos procesadores dedicados que son generalmente SIMD). Hoy en día esta clasificación no nos permite distinguir entre todas las clases de ordenadores paralelos que existen e incluso es muy poco útil para las arquitecturas convencionales que han evolucionado lo suficiente como para que se pueda pensar en distinguir diversas categorías.

3.2 Otros tipos de taxonomía

Como hemos puesto de relieve a lo largo de todos los capítulos anteriores, una estructura muy común que aparece al tratar con sistemas complejos es la jerarquía. Gran parte de las clasificaciones, de arquitecturas de ordenadores o de cualquier otro tipo, se caracterizan por su naturaleza eminentemente jerárquica y se distinguen entre sí, las que operan sobre la misma clase de sistemas, por los niveles que consideran y los que detallan con mayor precisión.

Con los ordenadores sucede exactamente lo mismo. Partiendo de la clasificación de Flynn, que considera dos niveles, datos y control, se pueden detallar aún más esos niveles o se pueden tratar de identificar otros, superiores o inferiores, con lo que se llega a otra clasificación de los mismos sistemas. De hecho, mucha de las propuestas existentes parten del esquema de Flynn. Pero no es la única posibilidad, las clasificaciones tiene un objetivo muy claro, saber qué se ha conseguido y descubrir por dónde se puede avanzar de acuerdo con algún criterio que se convierte en el eje de la clasificación. Se puede pensar en clasificaciones que atienden al número de usuarios, a velocidad de proceso, al número de procesadores que existen, a prestaciones concretas, etc.

Skillicorn [1988, p. 47] recoge varias de estas clasificaciones que resumimos brevemente. T.Y. Feng, en 1972, propuso una clasificación orientada a las prestaciones en cuanto al paralelismo que permite la máquina. El paralelismo se evalúa en función del número de bits que se pueden procesar simultáneamente y se representa como un par de números, el primero la longitud de la palabra con que se trabaja y el segundo el número de palabras que se pueden procesar simultáneamente. Evidentemente, este esquema es muy útil para comparar muy diversas arquitecturas pero no permite resaltar las diferencias ni las similitudes ya que no se hace referencia al tamaño, número de procesadores, precio o aplicación.

Reddi y Feurstel, en 1976, propusieron una clasificación más descriptiva en la que las arquitecturas se clasifican de acuerdo con su organización física, el flujo de información y cómo se representa y transforma ésta. Siendo más genérica que la de Feng, estos parámetros están demasiado orientados a la implementación específica del ordenador pues muchas máquinas resultan ser prácticamente iguales con diferentes organizaciones físicas (por ejemplo, microprogramación vs. cableado). Händler, en un trabajo presentado en 1977, describe las arquitecturas dando el número de procesadores y cómo se interconectan, así como el tamaño de palabra y la profundidad de las unidades aritmético-lógicas. Esta última clasificación está orientada a las máquinas de proceso vectorial y es difícil de generalizar a arquitecturas multiprocesador.

Otras dos clasificaciones más, que detallamos a continuación, son las propuestas por Skillicorn [1988] y Dasgupta [1990], que son un ejemplo muy claro de refinamientos sucesivos de clasificaciones previas y de propuestas de notación, cuando menos curiosas. Skillicorn parte del esquema de Flynn, y Dasgupta desarrolla aún más la propuesta de Skillicorn dotándola de una gran potencia denotacional.

3.3 Una ampliación al trabajo de Flynn

En esta clasificación se opta por una aproximación al problema de la computación y las máquinas a través de varios niveles de abstracción. En el nivel más alto estaría el modelo de computación. Uno de los más conocidos es el de Von Neumann, pero existen muchos más como el de flujo de datos, el de reducción de grafos, los modelos de pilas, paralelos, etc. Un nivel con un grado mayor de refinamiento sería el de máquinas abstractas (sobre las que se implementa el modelo computacional) ya que cada modelo computacional se puede implementar sobre máquinas muy diferentes, aunque se ajuste mejor a determinadas estructuras. Este es el nivel más alto de la clasificación de Skillicorn.

El siguiente nivel sería el de la implementación, la máquina tal y como la ve el programador en lenguaje ensamblador (ésta es una de las definiciones clásicas de arquitectura de ordenadores). Este es el segundo nivel de la clasificación de Skillicorn. Como el mismo reconoce, hay parámetros importantes que deja fuera

de la clasificación, uno de ellos es el lenguaje de programación ya que hay "una creciente tendencia a construir conjuntamente el lenguaje y la máquina" (véase qué decíamos anteriormente sobre la integración hardware/software).

Los parámetros que describen el primer nivel de la clasificación son los siguientes:

- El número de procesadores de instrucciones (IP),
- el número de memorias de instrucciones (IM),
- el tipo de conmutadores que conectan las memorias de instrucciones y los procesadores de instrucciones,
- el número de procesadores de datos (DP),
- el número de memorias de datos (DM),
- el tipo de conmutadores que conectan los procesadores de datos y las memorias de datos,
- el tipo de conmutador que conecta los procesadores de instrucciones (IP) y los procesadores de datos (DP),
- el tipo de conmutador que conecta los procesadores de datos (DP) entre sí.

Este primer nivel extiende las clases de Flynn considerando algunas características importantes de las clasificaciones actuales, entre ellas, el tipo de conexión entre los diversos procesadores, un parámetro fundamental en la arquitectura y organización interna del ordenador y decisivo a la hora de considerar sus prestaciones. El segundo nivel considera la posibilidad de "pipeline" en los procesadores y el comportamiento de su diagrama de estados.

La siguiente tabla (figura 3) muestra una posible clasificación atendiendo únicamente al primer nivel y suponiendo que el número de memorias coincide con el número de procesadores:

Class	IPs	DPs	IP-DP	IP-IM	DP-DM	DP-DP	Name
1	0	1	none	none	1-1	none	reduct/dataflow uniprocessor
2	0	n	none	none	n-n	none	separate machines
3	0	n	none	none	n-n	n x n	loosely coupled reduct/dataflow
4	0	n	none	none	n x n	none	tightly coupled reduct/dataflow
5	0	n	none	none	n x n	n x n	
6	1	1	1-1	1-1	1-1	none	von Neumann uniprocessor
7	1	n	1-n	1-1	n-n	none	
8	1	n	1-n	1-1	n-n	n x n	Type 1 array processor
9	1	n	1-n	1-1	n x n	none	Type 2 array processor
10	1	n	1-n	1-1	n x n	n x n	
11	n	1	1-n	n-n	1-1	none	
12	n	1	1-n	n x n	1-1	none	
13	n	n	n-n	n-n	n-n	none	separate von Neumann uniprocessors
14	n	n	n-n	n-n	n-n	n x n	loosely coupled von Neumann
15	n	n	n-n	n-n	n x n	none	tightly coupled von Neumann
16	n	n	n-n	n-n	n x n	n x n	
17	n	n	n-n	n x n	n-n	none	
18	n	n	n-n	n x n	n-n	n x n	
19	n	n	n-n	n x n	n x n	none	Denelcor Heterogeneous Element Processor
20	n	n	n-n	n x n	n x n	n x n	
21	n	n	n x n	n-n	n-n	none	
22	n	n	n x n	n-n	n-n	n x n	
23	n	n	n x n	n-n	n x n	none	
24	n	n	n x n	n-n	n x n	n x n	
25	n	n	n x n	n x n	n-n	none	
26	n	n	n x n	n x n	n-n	n x n	
27	n	n	n x n	n x n	n x n	none	
28	n	n	n x n	n x n	n x n	n x n	

Fig. 3. Cuadro con la clasificación de arquitecturas de Skillicorn

3.4 Química, arquitectura y ordenadores

Esta clasificación es una mejora explícita de la propuesta de Skillicorn que intenta superar una serie de limitaciones que considera importantes. La primera de ellas es la falta de poder de predicción ya que no permite comparar arquitecturas que pertenecen a clases diferentes o cómo varían diferentes modelos de la misma línea arquitectónica. A pesar de que Skillicorn hace referencia explícita a la necesidad de un modelo jerárquico, Dasgupta sostiene que no lo es y critica que un concepto tan importante como el "pipeline" aparezca en un nivel diferente a las memorias y los procesadores.

El artículo de Dasgupta comienza haciendo un estudio de la ciencia de la taxonomía (clasificación) y enumera las propiedades que debe reunir una buena clasificación. De acuerdo con éstas, considera que la propuesta de Skillicorn carece de poder de explicación (aunque está latente), que no es un esquema jerárquico y que se olvida de aspectos importantes como el ya comentado o la diferencia entre memorias principales y memorias cache.

Para resolver estos problemas, Dasgupta propone una jerarquía basada en los modelos utilizados en la Química, utilizando profusamente términos como radicales atómicos, moléculas, radicales complejos, etc. Las entidades básicas, o átomos, son las siguientes:

- (iM) "interleaved Memory", memorias en las que se puede acceder a varias unidades básicas en un único ciclo de memoria,
- (sM) "single Memory", memorias simples, sean de datos o de instrucciones,
- (C) "cache", memorias "buffer" de acceso rápido,
- (sI) unidad simple (de un sólo paso o de una sola instrucción) de preparación de las instrucciones,
- (pI) unidad con "pipeline" de preparación de instrucciones,
- (sX) unidad simple (de una instrucción o de un solo paso) de ejecución de instrucciones,
- (pX) unidad con "pipeline" de ejecución de instrucciones.

Formula	Structure
$C.sI$	
$(C.pX)_n$	
$C_m.sX_n$	
$C.(C_2.pl)_2$	
$iM_m.(C.pl)_n$	
$(sM.pl)(sM.C.pX_4)$	

Fig.4. Ejemplos de "formulación" de arquitecturas

Si una máquina tiene varios átomos, se denota con un subíndice añadido a los átomos (por ejemplo, iM_3 o pX_4). También se pueden construir radicales compuestos de varios átomos, por ejemplo: $(iM)_m.(C.pl)_n$, que significa m memorias del tipo iM , asociadas a n procesadores de instrucciones con "pipeline" y dotados, cada uno de ellos, de memoria cache. Una molécula representa un subsistema completo dentro de un ordenador dado, puede ser una molécula X (de ejecución) o I (de preparación de las instrucciones). Una macromolécula combina una molécula I y una molécula X y representa un ordenador completo. Sobre esta estructura se definen una serie de reglas sintácticas para combinar todos los elementos y formular las expresiones adecuadas a cada caso. En la figura 4 se recogen algunos ejemplos de esta clasificación.

Esta clasificación de Dasgupta es un buen ejemplo de cómo aprovechar notaciones ya aceptadas en otros campos para desarrollar una propia y cómo pueden utilizarse las analogías para facilitar la comprensión y el manejo intuitivo de la taxonomía. Sin embargo, como el propio Dasgupta reconoce, tanto esta clasificación como la de Skillicorn están seriamente limitadas cuando se trata de arquitecturas que no siguen el modelo Von Neumann.

4. Ordenadores personales y estaciones de trabajo

Estudiar las diferentes arquitecturas de ordenador es una tarea muy complicada dada la gran variedad de posibilidades existentes. Pero hay algo aún más difícil, clasificar los ordenadores que están disponibles comercialmente, no según el número de procesadores o de memorias, sino siguiendo la denominación tradicional de Ordenador Personal (pc), Microordenador, Estación de Trabajo (Workstation), Miniordenador, Superordenador, etc.

¿Qué significa cada una de estas categorías?, evidentemente nadie lo sabe con exactitud. En primer lugar, las definiciones de cada una de ellas abundan casi tanto como los productos comerciales y, en segundo lugar, la falta de rigor es casi absoluta. Al observar el panorama existente en las publicaciones especializadas cabe preguntarse si la borrosidad que todo el mundo parece admitir en este sentido no es más que un intento fallido de reducir por las bravas la complejidad de descripción de la oferta informática.

Aquí nos vamos a centrar en lo que parecen ser los dos escalones más bajos, los pc's y las Estaciones de Trabajo (WS, a partir de ahora). Con ellos se puede seguir mejor que con ningún otro el vertiginoso desarrollo de los ordenadores, los cambios que en ellos se han producido y las consecuencias que tuvieron y que tienen a la hora de concebir la informática. Por otro lado, son la referencia más inmediata que tienen la gran mayoría de los usuarios, alejados de los grandes ordenadores y las máquinas especializadas en aplicaciones científicas muy concretas.

4.1 Su majestad el pc

Resumir, aún concisa y brevemente, la historia del Ordenador Personal sería tan sólo repetir lo que otros muchos han hecho en multitud de libros y artículos. Y a pesar de que también existen análisis de todos los tipos, vamos a detenernos brevemente con algunas conclusiones que se pueden sacar de esa historia.

Definir un Ordenador Personal es muy arriesgado, entre otras cosas porque es muy probable que la definición resulte superada al cabo de muy poco tiempo. Sin embargo, parece que existe una serie de características que sin llegar a constituir una definición clara permiten diferenciarlo de alguna manera de otro tipo de ordenadores. Dada la creciente y fuerte tendencia a la normalización, un pc generalmente trabaja bajo un sistema operativo interactivo (el más extendido es el DOS, de Microsoft), es monousuario, con una memoria que oscila entre los 250 Kb y 4 Mb, con una o dos unidades de disco flexible, y posiblemente con un disco duro de varias decenas de Mb. Es posible conectarlos en red, disponen de varias ranuras de ampliación y el tipo de monitores suele ser de 80x25 columnas con diversas resoluciones en modo gráfico, siempre limitadas por el tamaño del monitor.

Los ordenadores personales y la complejidad

La principal ventaja que ofrecen los ordenadores personales es que ponen al alcance de cualquiera una potencia de cálculo impensable no hace mucho tiempo. La ciencia de hace un siglo se encontraba fuertemente limitada por la falta de instrumentos de cálculo y sólo podía tratar con problemas de una complejidad abordable que, para los estándares de la época, no era muy elevada (recordar el capítulo dedicado a la simplificación y el ejemplo del estudio del sistema solar).

Evidentemente, el H y el O eran exactamente el mismo ahora que hace un siglo pero el I ha cambiado radicalmente y, en consecuencia, también ha cambiado mucho la Imagen del Objeto que nos hemos formado. Al aparecer los primeros ordenadores se pudo empezar a tratar con problemas de una complejidad mucho mayor pero esta posibilidad sólo estaba al alcance de unos pocos privilegiados. Poco a poco fue haciéndose más fácil acceder a los ordenadores hasta que empezaron a aparecer los primeros Ordenadores Personales que brindaron esa posibilidad a todo el mundo. Es decir, permitieron que cambiara la IO de mucha más gente.

La lista de descubrimientos científicos de mayor o menor importancia que se han hecho utilizando un ordenador personal es enorme ya que permite que mucha más gente investigue y estudie un determinado objeto, con lo que las posibilidades de profundizar en el conocimiento sobre ese objeto son mucho mayores. Antes no todo el mundo disponía del tiempo o de la capacidad necesaria para resolver, por ejemplo, un conjunto complicado de ecuaciones, ahora éste es un trabajo que realiza el ordenador y que además lo realiza de forma más precisa y permitiendo ampliar detalles que antes no se podían considerar (de nuevo nos remitimos al ejemplo del sistema solar que estudiamos en el capítulo dedicado al tratamiento de la complejidad). Desde luego siempre están los superordenadores como las herramientas dedicadas a tratar con problemas de una complejidad enorme, pero los Ordenadores Personales, aún disponiendo de una capacidad incomparablemente menor, permiten que cada usuario individual maneje una complejidad considerable con un coste muy reducido.

En este apartado dedicado a los ordenadores personales vamos a estudiarlos desde el punto de vista de su evolución en el tiempo y la creciente expansión en el mercado tanto de los ordenadores personales como de las estaciones de trabajo. Lo que aquí digamos ha de interpretarse de acuerdo con la idea de que los ordenadores son básicamente herramientas para tratar la complejidad y que los Ordenadores Personales son herramientas que nos permiten tratar la complejidad de muchos objetos cotidianos.

Evidentemente todas estas características son discutibles y lo serán cada vez más a medida que pase el tiempo, pero aún así pueden dar una idea bastante clara de lo que es un pc.

La gran ventaja que ofrecen es poner a disposición del usuario no especializado una enorme capacidad de proceso unida a una flexibilidad que no tiene prácticamente ningún otro tipo de ordenador. Si cuando aparecieron ya se consideraron como un hecho "revolucionario" en la forma de trabajar, esa revolución continúa pues los avances han permitido multiplicar por un factor muy elevado todas las prestaciones que ofrecen.

El ordenador es una herramienta que permite manejar mucha complejidad. Esto es algo que hemos venido repitiendo con frecuencia a lo largo de estos apuntes. El ordenador personal es una herramienta accesible a mucha gente a la que le permite gestionar de forma individual problemas cada vez más complejos. Los primitivos ordenadores personales con 64 Kb de memoria y procesadores de 8 bits permitieron que muchos científicos, por citar algún ejemplo concreto, dispusieran de una herramienta personal (y este adjetivo es lo importante) para abordar los problemas que estudiaban. Muchos de los estudios realizados en la segunda mitad de este siglo son inseparables de la historia del ordenador y no se podrían haber abordado sin éstos. Un ejemplo muy claro es la dinámica de sistemas cuyo potencial sólo es aprovechable si se dispone de un ordenador capaz de ejecutar rápidamente todas las simulaciones que sean necesarias. Y lo mismo sucede con la predicción del tiempo atmosférico, el estudio del caos, los fractales, estudios aerodinámicos, controles de centrales de energía y un largo etcétera.

Pero la mayoría de esas aplicaciones requerían y requieren grandes ordenadores manejados por especialistas, la importancia del pc es que puso una parte de ese poder de computación en manos de los usuarios. Hoy en día existen ordenadores que aún estando dentro de la categoría de los pc permiten realizar complicadas tareas de diseño y análisis que no hace mucho eran competencia exclusiva de los grandes ordenadores. Pero aún dejando de lado esas aplicaciones especializadas, los pc's permiten que cualquiera, cómodamente instalado en su casa, realice complejos análisis financieros, prepare documentos de gran calidad, lleve la contabilidad de pequeñas e incluso medianas empresas, utilice sistemas expertos, programas de análisis de circuitos y un sinfín de tareas que antes eran inabordables.

Además, la enorme presión comercial en este mercado hace que los precios bajen cada vez más y las máquinas sean cada vez más potentes. Cualquiera familiarizado con el mundo de los microprocesadores conoce la saga de los 8086, 8088, 80186, 80286, 80386 y 80486, todos ellos aportando verdaderos saltos cualitativos en prestaciones y concepción de los ordenadores personales. Modelos como el 8 Mb Mac II de Apple, PS/2 Modelo 80 de IBM o el COMPAQ Deskpro 386/40 tienen memorias de alrededor de 10 Mb y procesadores que funcionan a 16,7 MHz con

capacidad multitarea, a un precio que en ningún caso llega a los 10.000 \$ (y algunos son bastante más baratos).

Por otro lado, la enorme flexibilidad con que están diseñados permiten convertir a un ordenador personal en un auténtico ordenador especializado que pone a disposición de los usuarios un potente centro de cálculo. Personalizar un pc para una aplicación concreta es probablemente más barato que comprar un ordenador grande capaz de abordar ese mismo problema directamente. Existen tarjetas de ampliación con procesadores especializados en proceso numérico, procesado de imágenes, procesado de voz, arrays sistólicos, tarjetas de adquisición de datos, periféricos para programar PLD's y memorias, posibilidades de conectarse a todo tipo de redes, convertir un pc en un banco de pruebas de circuitos integrados, controladores de equipos con pantallas táctiles y multitud de periféricos que permiten hacer de un ordenador personal una potente máquina especializada.

4.2 Estaciones de trabajo

Como su propio nombre indica, las estaciones de trabajo fueron inicialmente concebidas para aplicaciones muy especializadas que requerían gran potencia de cálculo y capacidad de memoria. El desarrollo de este tipo de máquinas es, de alguna manera, paralelo al de los ordenadores personales pero aplicado a la ciencia y a la ingeniería.

Definir una WS es tan complicado como definir un ordenador personal, normalmente se entiende como tal una máquina monousuario, con capacidad de multitarea y, como mínimo, prestaciones de 1 MIPS (Mega Instructions Per Second), conectable mediante red Ethernet, sistema Operativo UNIX, aplicaciones técnicas y de ingeniería y monitor de alta resolución.

Aplicaciones típicas de las estaciones de trabajo son programas de CAE/CAD (Diseño Asistido por Ordenador) y la mayor parte de los usuarios de estos ordenadores son técnicos y especialistas en diversas ramas de la ciencia y la ingeniería. Son, en general, problemas que podrían resolverse con un pc pero a costa de pagar un alto precio en eficiencia y velocidad y con limitaciones importantes (sobre todo en procesamiento numérico y representación gráfica), de hecho muchas de las aplicaciones existentes en el mercado de WS también existen en el mercado de los ordenadores personales aunque en versión reducida. Ya hemos mencionado el diseño asistido por ordenador (que necesita una alta resolución gráfica y unas prestaciones numéricas importantes para procesar las imágenes) pero también conviene mencionar el diseño industrial (de componentes mecánicos y electrónicos y las correspondientes pruebas y análisis), modelado de elementos finitos, animación de vídeo, análisis térmicos y electromagnéticos, estudios de dinámica de fluidos, programación de grandes sistemas, fabricación asistida, etcétera.

Una idea aproximada de la potencia de las actuales estaciones de trabajo es la siguiente: "con los sistemas actuales, los ingenieros pueden diseñar, por ejemplo, una pieza del ala de un avión, con la próxima generación de ordenadores más potentes podrán diseñar el ala entera y con futuras generaciones serán capaces de diseñar el avión completo" [LoPiccolo, 1988].

Sin embargo existe un problema que ya hemos comentado en otros apartados de este capítulo. El hardware parece ir siempre muy por delante del software y, aunque la tecnología y los ordenadores cada vez son más potentes, no se dispone de las herramientas necesarias para sacarle partido. Y hay muchos factores que contribuyen a aumentar aún más ese distanciamiento entre el hardware y el software, las máquinas RISC (ver el apartado dedicado a este tema en este mismo capítulo) han permitido aumentar de forma importante la potencia y prestaciones de las estaciones de trabajo, lo mismo sucede con el paralelismo (especialmente las arquitecturas como los *arrays* sistólicos) que permite realizar simultáneamente muchas operaciones que antes se tenían que hacer de forma secuencial. La tecnología de las estaciones de trabajo es tan fluida que es muy difícil adaptar el software existente a las nuevas máquinas que van saliendo al mercado y más si estas aparecen a la velocidad con que lo hacen.

Al contrario de lo que sucede en los pc's, donde las prestaciones son importantes pero lo es aún más disponer de un programa para aprovechar la flexibilidad que ofrecen, en las estaciones de trabajo el parámetro fundamental es la velocidad y la capacidad, siempre se necesita más potencia de cálculo y mayor resolución gráfica. A estas demandas responden la industria y la investigación hardware pero la velocidad de desarrollo hace muy difícil disponer del software adecuado y más si tenemos en cuenta que las estaciones de trabajo suelen utilizarse de forma muy especializada (para una o dos aplicaciones fundamentales).

Quién es quién en el mercado de las estaciones de trabajo

Hay cinco grandes compañías que compiten por el mercado de las estaciones de trabajo (4 si tenemos en cuenta que Hewlett-Packard compró Apollo). Apollo Computer Inc., ofrecía tres familias de estaciones: 3000, 4000 y 5XX, estas dos últimas de altas prestaciones. Sun Microsystems Inc. (el nombre Sun significa Stanford University) ofrece una gama muy amplia de estaciones de trabajo con unas prestaciones muy interesantes gracias a sistemas muy novedosos como la estación Sun 4/200, con arquitectura RISC o el sistema distribuido de ficheros NFS, que permite trabajar con unidades sin disco al compartirlos a través de la red. Hewlett-Packard, conocida como la compañía de ingenieros para los ingenieros, presenta una línea de estaciones en las que la arquitectura RISC también juega un papel muy importante. Digital Equipment Corporation (DEC) tiene la gran ventaja de ser uno de los líderes mundiales en redes de ordenadores y ofrecer una gran compatibilidad entre máquinas y un parque instalado muy amplio (los conocidos VAX, a los que las estaciones de trabajo, Micro VAX, se pueden conectar de forma sencilla algo que no es cierto en la mayoría de los otros equipos). IBM, el gigante azul, fue la primera compañía que anunció una estación de trabajo con arquitectura RISC, el IBM RT PC, que, sin embargo, no fue un éxito comercial. Los nuevos sistemas PS/2 y el nuevo modelo de estación, 9370, intentan recuperar ese terreno perdido. Pero este panorama está cambiando continuamente, desde luego a mayor velocidad de la que rige el ritmo de una publicación como ésta.

Existen teorías que aseguran que el número de errores que comete un ingeniero al diseñar un dispositivo determinado decrece de una forma muy importante con el aumento de la velocidad de la respuesta del ordenador, debido a que no tiene que esperar esta respuesta y no se interrumpen sus procesos mentales [Stenzel, citado por Lopiccio, 1988]. Evidentemente, si se está realizando un diseño gráfico, representa una gran ventaja ver inmediatamente cuáles son las consecuencias de una acción determinada (esto, dada la complejidad de las tareas involucradas es algo que no se puede conseguir con un pc y quizá sea un tema interesante a estudiar a la hora de diseñar las interfaces de las aplicaciones de los pc's). Lo mismo se puede decir de la resolución gráfica, se puede entender mucho mejor lo que se está haciendo si se ve correctamente y con detalle (algo difícil de conseguir con los pc's).

Algunas medidas de productividad aseguran que la productividad de los ingenieros puede incrementarse en un 30-40 % si utilizan una estación de trabajo. Este dato es muy importante, porque el primer objetivo de las estaciones de trabajo es conseguir importantes beneficios de la inversión, como pueden ser mejoras en el proceso de diseño o ciclos de diseño y fabricación más cortos.

4.3 Ordenadores personales vs. estaciones de trabajo

El principal problema que se le plantea a un usuario es decidir qué es lo que necesita para sus aplicaciones, un ordenador personal o una estación de trabajo. Realmente no es fácil de decir aunque los productores de este tipo de equipos parecen tener muy clara la diferencia entre ambos mercados. No cabe ninguna duda de que los avances en la tecnología de los pc's hacen que éstos sean cada vez más parecidos a las estaciones de trabajo. Los microprocesadores utilizados en modelos de ordenador personal comerciales y ampliamente extendidos ya permiten multitarea, manejan varios megas de memoria y utilizan dispositivos gráficos de alta resolución, las prestaciones son buenas y se pueden mejorar de forma importante si se recurre a la personalización del equipo. También son capaces de funcionar bajo el sistema operativo UNIX intentando aprovechar las aplicaciones escritas para este sistema. Pero las estaciones de trabajo evolucionan a la misma velocidad que los ordenadores personales y sus prestaciones mejoran a gran velocidad. Una forma de interpretar este dilema es considerar que la gama alta (los mejores equipos de los pc's, con mayores prestaciones y más caros) puede llegar a confundirse con la gama baja de las estaciones de trabajo (las no muy especializadas y con prestaciones "normales" dentro de los márgenes que manejan las WS). Algunos llegan a hablar incluso de las pc-Workstations en contraposición a las "Technical Workstations" para diferenciar entre unos y otros.

La figura 5 muestra las prestaciones de tres equipos punteros dentro de la oferta de los ordenadores personales en un momento histórico ya claramente sobrepasado.

Memory Capacity	8	16	10
MB/ board - vendor	1	2	n/a
MB/board - Clearpoint	1	n/a	n/a
Bus Architecture	NuBus	MicroChannel	Modified AT
Expansion Slots (memory)	8	8	n/a
Error Checking	none	Parity	Parity
Processor	16.7 MHz 68020	16.7 MHz 80386	16.7 MHz 80386
Math Co-processor	68881	80387(opt)	80387 or Weitek
MIPS	2	1	n/a
Operating System	Apple, A/UX	OS/2(1988)	OS/2 (1988)
*Base System Price	4300	6995	7999

Fig. 5. La gama alta de los ordenadores personales [Engineering Tools, 1988, p. 19]

Si comparamos los precios de estos equipos con los que aparecen en el cuadro de las estaciones de trabajo llegamos a una de las conclusiones más importantes a la hora de decidir entre un pc o una WS. Los pc's no admiten utilizar UNIX de forma natural y son más difíciles de conectar en red (un aspecto fundamental en toda instalación) y sus prestaciones no justifican su precio en muchos casos. Aún así, el número de aplicaciones que existen para WS no es comparable a la ingente cantidad de programas escritos para pc, si lo que se desea es flexibilidad y un espectro amplio de utilización la balanza se inclina definitivamente por los pc's (eso sin tener en cuenta la diferencia de precio entre el software para pc y el software para WS).

La figura 6 recoge las prestaciones y precio de algunas estaciones de trabajo agrupadas por compañías y en las diferentes series existentes. Es importante comparar estos datos con los del cuadro anterior para tener una idea más clara de dónde están las diferencias.

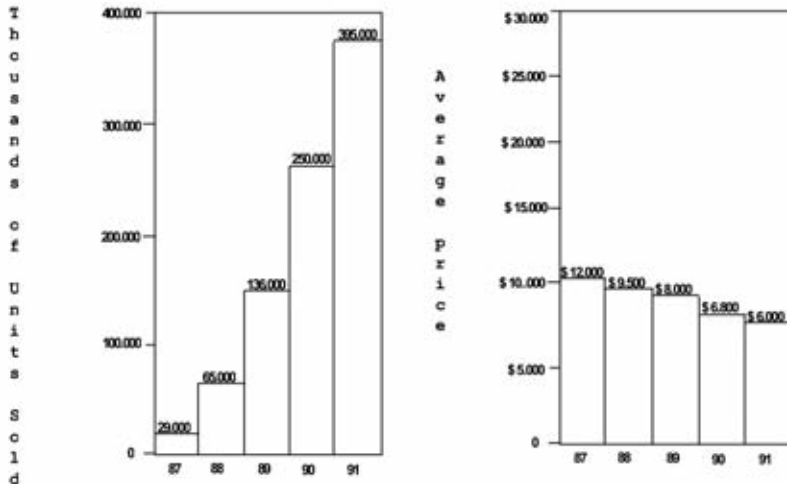
Memory Capacity	8	32	32	16	6	32
MB/ board - vendor	1,2	8	8	8	4	8
MB/board - Clearpoint	1,2	8	na	16	16	na
Bus Architecture	PC-AT	PC-AT	VME	Q-Bus	none	Q-Bus
Expansion Slots(memory)	4	4	4	2	none	4
Error Checking	Parity	Parity	EDC	Parity	Parity	EDC
Processor	12 MHz	25 MHz 68020	20 MHz	20 MHz	20 MHz	22 MHz Micro VAX
Math Co-processor		68881		78132		CMOS FPU
MIPS	1,5	4	n/a	9	9	3
Operating System	Domain/IX	(UNIX clone)		VMS, Ultrix (UNIX clone)		
*Base System Price (\$)	4990	13,900	46,900	26,000	5400	50,400

Fig. 6. Cuadro de prestaciones y precios de algunas estaciones de trabajo

Pero apenas se ha definido una situación, ésta cambia con una velocidad sorprendente. Al igual que sucedió con los pc's, las aplicaciones de las WS de carácter más técnico están dando paso a otras mucho más genéricas (existe un estudio en el que se afirma que el 70% de utilización de las estaciones de trabajo se dedica a funciones administrativas). De alguna manera, las estaciones de trabajo están invadiendo el terreno de los pc's. Y como todo lo que pueda ser un mercado es aprovechable, están empezando a aparecer aplicaciones de proceso de texto, hojas electrónicas, publicación electrónica y bases de datos con interfaces mucho más conviviales que trasladan una tarea clásica del ordenador personal a las más potentes estaciones de trabajo. En la figura 7 se muestran unas expectativas

de desarrollo del mercado tanto de las estaciones de trabajo como de los pc's (nótese la agrupación que antes mencionábamos en pc-Workstations y Technical Workstations).

PC-WORKSTATIONS



TECHNICAL WORKSTATIONS

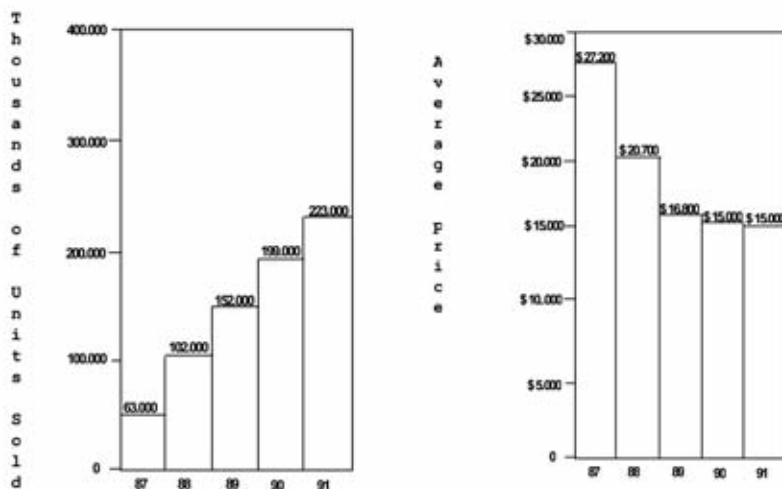


Fig. 7. Evolución del mercado de ordenadores personales y estaciones de trabajo [Engineering Tools, 1988, p. 87]

5. RISC, la simplificación del diseño

A partir de la propuesta inicial de Von Neumann y gracias a los avances en electrónica, se fueron construyendo ordenadores cada vez más complejos. Las primeras máquinas no eran más que un conjunto de puertas lógicas con una arquitectura muy simple, que se fue complicando a medida que aumentaba la capacidad de integración de circuitos. Para ganar en velocidad y eficiencia se fueron implementando en hardware cada vez más niveles de la jerarquía y la potencia de computación del hardware se multiplicó por factores muy elevados.

Esto dió lugar a arquitecturas extremadamente complejas, con un lenguaje máquina de varios cientos de instrucciones que permitían realizar prácticamente cualquier operación imaginable (por ejemplo, el VAX-11 tiene instrucciones máquina directamente orientadas a lenguajes de alto nivel como son INDEX, para utilizar matrices, CASE, para la estructura de alto nivel del mismo nombre o CALL, para llamadas a procedimientos con paso de parámetros). Muchas de las instrucciones de código máquina eran incluso más potentes que algunas instrucciones de lenguajes de alto nivel. Como consecuencia, la complejidad organizativa de los procesadores y elementos asociados creció considerablemente. Las razones para construir conjuntos de instrucciones complejos eran dos fundamentalmente [Sáez Vacas, 1987, p. 60]:

- Aproximar los niveles semánticos entre varios lenguajes de alto nivel y una parte de los niveles de lenguaje ensamblador y máquina. Al implementar en hardware instrucciones muy complejas se permite que instrucciones de alto nivel puedan ejecutarse casi sin necesidad de ser traducidas por el compilador. De forma extrema, se puede pensar en un lenguaje máquina que coincide con el lenguaje de alto nivel.
- Mejorar la relación procesador/memoria, al reducirse el número de accesos a memoria. Una instrucción compleja permite operar mucho más sobre un dato determinado sin necesidad de traerlo y llevarlo a memoria.

Esto está en la línea que ya comentábamos al principio del capítulo al hablar sobre integración de hardware y software.

Sorprendentemente, era ampliamente conocido y estaba suficientemente documentado que los programadores y, en consecuencia, los compiladores (que son los que más frecuentemente acceden al código máquina), utilizan un conjunto bastante reducido de instrucciones que supera el 75 % de frecuencia de utilización de las instrucciones del lenguaje [Sáez Vacas, 1987, p. 59].

Para reducir la complejidad de diseño se inició a finales de los 70 y principios de los 80 una tendencia a diseñar las arquitecturas con un conjunto reducido y simple de instrucciones, comenzando así la singladura de los ordenadores RISC (Reduced Instruction Set Computers).

Tempus y Hora, una metáfora RISC

En el capítulo dedicado al tratamiento de la complejidad estudiábamos la propuesta de simplificación que, en forma de parábola, hace H.A. Simon. El problema que plantean las arquitecturas RISC es muy similar. Los conjuntos complejos de instrucciones consumen muchos ciclos de reloj por instrucción, de forma que cada vez que aparece una interrupción el sistema o salva el estado global (muy complejo, dada la estructura del procesador) o desperdicia mucho trabajo reiniciando la instrucción interrumpida. La aproximación RISC propone instrucciones mucho más simples (teóricamente de un único ciclo de reloj de duración) de forma que las interrupciones no suponen un gran problema y, cuando se necesitan operaciones complicadas (que como comentábamos antes es menos frecuente de lo que a primera vista pudiera parecer) se pueden construir a partir de las más simples. El esfuerzo se traslada a la compilación, pudiéndose así construir procesadores mucho más simples, eficientes y rápidos.

La filosofía de diseño RISC plantea las siguientes premisas [Gimarc, 1987, p. 59]:

1. Analizar las aplicaciones para determinar qué operaciones se utilizan con más frecuencia.
2. Optimizar la ruta de datos para ejecutar esas instrucciones tan rápidamente como sea posible.
3. Incluir otras instrucciones sólo si se ajustan a la ruta de datos previamente especificada, son relativamente frecuentes o su inclusión no ralentiza la ejecución de las instrucciones más frecuentes.
4. Aplicar la misma filosofía a todos los recursos del procesador, incluir sólo los más frecuentemente utilizados y sólo si no ralentizan recursos más utilizados.
5. Trasladar toda la complejidad que sea razonablemente posible del hardware de ejecución al software de compilación.

A partir de ellas, las máquinas RISC ofrecen una serie de características comunes que todas comparten en mayor o menor grado: [Gimarc, 1988, p. 59]:

- Ejecución de la mayoría de las instrucciones en un solo ciclo.
- Conjunto de instrucciones de recuperación y almacenamiento.
- Decodificación cableada de instrucciones.

- Relativamente pocas instrucciones y modos de direccionamiento.
- Formato de instrucciones fijo para facilitar la decodificación.
- La complejidad se traslada a compiladores optimizados.
- Ruta de datos muy segmentada para facilitar concurrencia.
- Conjunto amplio de registros.
- Múltiples niveles de memoria.
- Conjunto de instrucciones diseñado para un grupo determinado de aplicaciones.

La controversia entre ordenadores RISC y CISC puede dividirse en dos categorías: cómo diferenciar un ordenador RISC de uno CISC y cómo hacer comparaciones razonables entre ellos.

MOTOROLA amplía su familia de microprocesadores CISC

(Mundo Electrónico, Abril 1990, p. 51)

"El nuevo miembro de la familia 680X0 de microprocesadores CISC de 32 bits de Motorola dobla las prestaciones del 68030, integra más de 1,2 millones de transistores y consigue una media de ejecución sostenida de 1,3 instrucciones complejas por ciclo de reloj, valor muy próximo al teórico de 1 instrucción por ciclo de los microprocesadores RISC".

Este microprocesador, llamado 68040, integra en un único chip una unidad de cálculo de coma flotante y dos antememorias de 4Kbytes y se prepara en tres versiones, de 25, 33 y 50 MHz. Con técnicas de segmentación, puede realizar hasta 14 tareas simultáneas, la unidad de cálculo de enteros realiza 6 operaciones al tiempo y la unidad de coma flotante 3 operaciones simultáneas. La unidad de cálculo puede trabajar de forma independiente sobre las dos antememorias. Para evitar tener que salvar el estado del procesador cada vez que se produce una interrupción (operación muy compleja dada la cantidad de parámetros involucrados) sólo se almacenan los datos de partida y si se produce alguna interrupción, se reanuda la tarea desde el principio.

Hewlett-Packard anuncia una amplia oferta de productos basados en RISC

(Mundo Electrónico, Mayo 1990, p. 60)

"... con esta presentación de nuevos equipos, la mayor que realiza la compañía en sus 50 años de historia, H-P ofrece ahora la más amplia

oferta de ordenadores basados en la arquitectura RISC de todo el mercado". Hewlett-Packard es una de las compañías que más firmemente se ha comprometido con la arquitectura RISC con su línea de procesadores Spectrum (HP Precision Architecture) orientada a aplicaciones UNIX.

IBM anuncia la familia RISC System/6000

(Mundo Electrónico, Abril 1990, p. 56)

"IBM España ha presentado recientemente su nueva familia RISC System/6000 ...Estos nuevos equipos, que han supuesto una inversión de 1000 millones de dólares, mejoran la tecnología RISC, al tiempo que incorporan la nueva versión del sistema operativo UNIX de IBM, el AIX V.3". "Todos estos modelos utilizan la arquitectura POWER (Performance Optimization With Enhanced RISC), con un nuevo procesador super-escalar capaz de ejecutar varias instrucciones en un solo ciclo."

Desde el punto de vista conceptual, los ordenadores RISC son una aproximación simplificadora al diseño de arquitecturas de ordenadores partiendo de una filosofía coherente con lo que se quiere y lo que se necesita. Desde el punto de vista de la complejidad, constituyen un ejemplo muy interesante de reducción de la complejidad de diseño y un punto de partida a considerar al plantearse nuevas aproximaciones a los ordenadores, pero desde luego han de compararse cuidadosamente con arquitecturas tradicionales y no es tarea fácil. Desde el punto de vista tecnológico son una elección cuyas motivaciones y consecuencias cambian a gran velocidad, muchas de las características de los ordenadores RISC provienen de los CISC, como la ruta de datos segmentada o las memorias cache. Y a pesar de que fueron inicialmente pensados para tipos concretos de aplicaciones, hay una clara tendencia a ensanchar el espectro de aplicaciones y hacerlas similares en utilidad a los CISC. Todo ello conduce a una situación de confusión bastante extendida sobre las ventajas de unas y de otras arquitecturas, confusión que aumenta a medida que la tecnología reduce las diferencias que inicialmente eran más evidentes. Como prueba de ello mostramos el cuadro que recoge diversas noticias referentes a la presentación de diversos equipos informáticos.

6. Paralelismo

La enorme velocidad de evolución de los ordenadores ha permitido abordar problemas cada vez más complejos a través del aumento de la capacidad de computación. Pero aunque esto fuera impensable hace pocos años, se está llegando al límite de esas posibilidades, dada la concepción tradicional de los ordenadores. La arquitectura von Neumann, la más conocida y utilizada, presenta varios problemas de cuellos de botella que hasta ahora se han ido resolviendo mejorando el hardware que la soportaba. Cuando hizo falta más velocidad se construyeron procesadores más rápidos, cuando faltó más capacidad se construyeron procesadores más potentes, para responder a la necesidad de manejar enormes cantidades de datos se perfeccionaron las memorias, su velocidad y su capacidad, pero incluso todas esas mejoras y avances tiene un límite si no se cambia la arquitectura del ordenador.

El límite teórico en arquitecturas clásicas viene impuesto por la velocidad de transmisión de los electrones dentro de los circuitos que componen el ordenador (una fracción de la velocidad de la luz). Una velocidad ciertamente elevada pero insuficiente para muchos problemas que se plantean hoy en día a los ordenadores.

Un ejemplo de este tipo de problemas es la simulación de tiempo atmosférico. Estas simulaciones se hacen dividiendo la zona de la atmósfera a estudiar en una rejilla tridimensional y estudiando la evolución de las condiciones climatológicas en cada una de las divisiones, luego se integran todos los resultados en una solución global. La resolución y la exactitud aumentan disminuyendo el tamaño de las divisiones (ver en el capítulo dedicado a los **conceptos relacionados con la complejidad** los problemas que plantea el estudio de la atmósfera). Estos cálculos, para que sean útiles en predicción, han de hacerse en tiempo real, es decir, la predicción del tiempo para mañana no puede tardar dos días.

Para solucionar este tipo de problemas se pensó hace ya tiempo en recurrir a arquitecturas paralelas, en las que la idea básica es que si un procesador ejecuta 1000 instrucciones por segundo, 100 procesadores ejecutarán 100.000 instrucciones por segundo. En la práctica esto no es así, como veremos, pero este tipo de arquitecturas no sólo permiten abordar problemas como los mencionados sino que además pueden tratarse de un forma totalmente distinta. El ejemplo del tiempo atmosférico ilustra muy bien este punto. Los cálculos para cada división de la rejilla son independientes entre sí, salvo las condiciones de contorno que se imponen unas a otras y las influencias entre ellas, luego se puede pensar en realizar el análisis de cada una de estas divisiones en un procesador y, utilizando un número suficiente de procesadores, ejecutar todas al mismo tiempo.

6.1 Arquitecturas paralelas

Las diversas categorías de ordenadores paralelos se diferencian entre sí en dos parámetros fundamentales. El primero de ellos es cómo se va a conseguir el paralelismo, es decir, qué estructuras son las que van a funcionar en paralelo. Una vez decidido este punto hay que precisar el método de interconexión entre los elementos paralelizables. Las comunicaciones son un problema crítico dentro de un ordenador paralelo ya que es necesario coordinar todos los procesos que se ejecutan concurrentemente, el tipo de solución adoptada caracteriza a la máquina tanto o más que la propia arquitectura.

Como se dijo antes, la idea básica del paralelismo es que si un procesador consigue unas prestaciones X , entonces es lógico pensar que n procesadores conseguirán unas prestaciones nX . Esto, por supuesto, en teoría, ya que la coordinación de esos n procesadores supone una sobrecarga muy importante y hay problemas que no se pueden resolver de forma eficiente con un ordenador paralelo. De acuerdo con esto, las interpretaciones que se pueden hacer, y se han hecho, del paralelismo son muy diversas y afectan a todos los niveles de organización del ordenador. Desde las máquinas para proceso numérico con múltiples unidades funcionales a los ordenadores con varios procesadores completos, hay un amplio abanico de posibilidades que prácticamente se ha experimentado por completo, existiendo muchos prototipos y productos comerciales.

Aquí vamos a dar sólo un breve repaso a las diversas arquitecturas básicas y a algunos métodos de conexión, intentando resaltar la complejidad que resuelven y la que generan.

Arrays sistólicos. Se basan en el hecho de que en muchas aplicaciones matemáticas hay un núcleo básico de operaciones que se están utilizando constantemente, por ejemplo, las multiplicaciones de matrices, soluciones de sistemas lineales o transformadas de Fourier. Con un número suficientemente amplio de elementos funcionales (pequeños procesadores o unidades aritmético-lógicas) y las conexiones adecuadas entre ellos se puede conseguir paralelizar una operación matemática compleja reduciendo en mucho los tiempos de cálculo. En la figura 8 aparece un array sistólico para la multiplicación de matrices $[A] \times [B] = [C]$. Los arrays sistólicos ejecutan las operaciones y la Entrada/Salida al mismo tiempo y no necesitan ningún tipo de control.

Evidentemente, sólo son útiles en determinadas operaciones matemáticas y rentables únicamente en determinadas aplicaciones de proceso numérico y necesitan un ordenador de propósito general que realice el resto de las funciones. La eficiencia conseguida es muy grande pues se diseña el hardware exactamente para el tipo de operación a realizar.

Procesadores en array (vectoriales). El principio en que se basan es prácticamente el mismo que el de los arrays sistólicos. Sin embargo, la aplicación es más genérica pues la operación a realizar sobre los datos se envía a todos los procesadores que la ejecutan sobre sus datos locales. Este tipo de arquitecturas se utiliza para problemas muy estructurados que requieren manejar un gran número de datos en forma matricial. Al contrario de lo que sucede con los arrays sistólicos, en los que el programador no necesita saber siquiera que están, la eficiencia de los procesadores en array depende en gran medida de la programación y el diseño de algoritmos adecuados. Existen máquinas con mas de 16.000 elementos de proceso conectados en forma de matriz utilizadas para procesar imágenes de satélites, simulación del tiempo atmosférico, estudios aerodinámicos, procesamiento de imágenes radar, etc. Este tipo de arquitecturas introduce una de las controversias características del paralelismo ya que ofrecen la posibilidad de una velocidad de proceso muy elevada pero con el coste adicional de una mayor complejidad en el diseño de los programas.

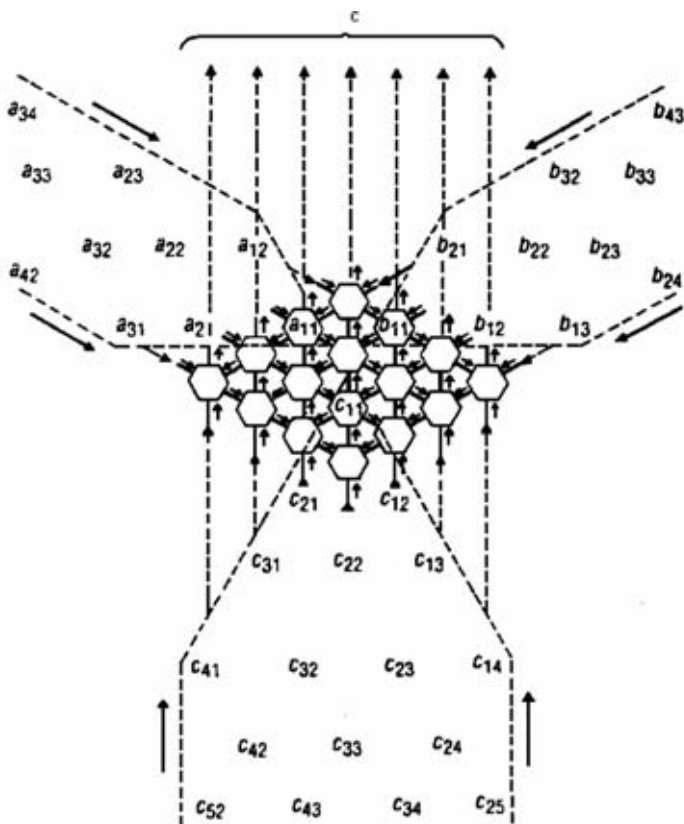


Fig. 8. Ejemplo de utilización de un array sistólico

Máquinas de flujo de datos. Este tipo de ordenadores propone una arquitectura radicalmente diferente a la de Von Neumann. En lugar de centrarse en las instrucciones (las instrucciones se ejecutan tras decodificarse y traer los datos correspondientes), en estas máquinas lo relevante son los datos, no existe nada parecido a un contador de programa o a una unidad de control. Para explicar mejor su funcionamiento utilizaremos la siguiente figura:

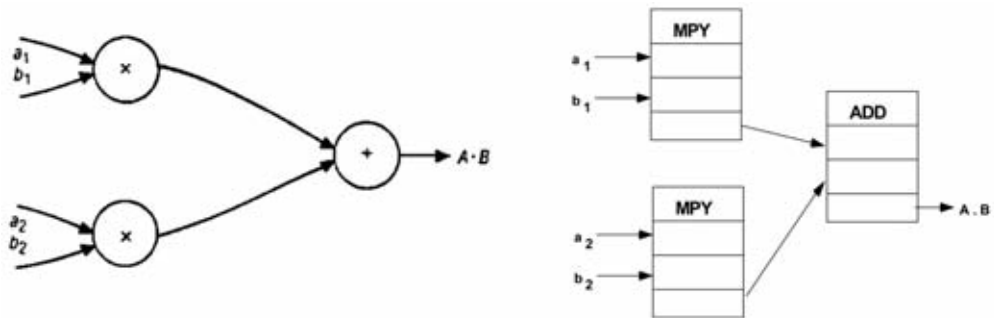


Fig. 9. Ejemplo de ejecución en una máquina de flujo de datos

Para realizar la operación $A \times B = a_1 \times b_1 + a_2 \times b_2$, cada operación a realizar se codifica en un patrón único (MPY y ADD, en este caso), cuando los datos están disponibles se pone la dirección del patrón en la cola de ejecución y se asigna un procesador para realizarla. Esto genera un paquete de resultados que sirve como dato para el siguiente patrón. Este tipo de máquinas plantea dos problemas importantes, los lenguajes de programación (muy diferentes de los tradicionales, complejos y con serias limitaciones) y la red de comunicación necesaria para llevar los datos de un procesador a otro, un aspecto crítico en todos los ordenadores paralelos. Una posible aplicación de estas arquitecturas es utilizarlas como unidades funcionales de un ordenador más general (al igual que sucede con los arrays sistólicos).

Máquinas de programación funcional. Este tipo de máquinas se han diseñado para ejecutar diversos tipos de lenguajes funcionales y aprovechar así su paralelismo inherente. La gran ventaja de estas máquinas es que una vez que el programador está familiarizado con los lenguajes de programación funcional no ha de preocuparse de especificar el paralelismo ya que éste es inherente al lenguaje y no hace falta un análisis profundo de la estructura del programa. Sin embargo, sí es tarea del programador escoger una solución paralela o paralelizable y descomponer el problema de la forma adecuada, tarea nada sencilla y que es objeto de un fuerte debate dentro de los que estudian el paralelismo, ya que muchos proponen que el paralelismo debería ser sólo en la máquina, el programador resuelve el problema como lo ha hecho tradicionalmente y es el compilador el encargado de paralelizar el programa.

Procesadores múltiples. La reducción del precio de los procesadores completos ha producido un interés creciente en las máquinas que incorporan varios procesadores como elementos básicos de bajo coste. En estos sistemas cada procesador se puede programar completamente y ejecutar su propio programa, de esta forma se consigue una flexibilidad mucho mayor que en las estructuras anteriores aunque esta flexibilidad se traduce en una complejidad mucho mayor en el control y la programación. Con este tipo de diseño se pueden conseguir factores de aceleración casi lineales pero existen una serie de factores a tener en cuenta: si los procesadores se han de coordinar cada cierto tiempo, las prestaciones se ven seriamente afectadas, existen muchos procesos que no son paralelizables y sólo se pueden ejecutar de forma secuencial, los algoritmos paralelos requieren más pasos que los secuenciales y una sobrecarga adicional para gestionar ese paralelismo, que varios procesadores compartan recursos puede ralentizar mucho determinados procesos y, finalmente, operaciones como la entrada/salida pueden llegar a consumir un tanto por ciento muy elevado del tiempo de computación reduciendo las ventajas del paralelismo.

Estos son los cinco tipos básicos de arquitecturas paralelas. Por supuesto, se podrían hacer muchas otras clasificaciones (tema que ya tratamos al hablar de las taxonomías) atendiendo a criterios diversos. Aparte de esta clasificación, la otra forma de estudiar los ordenadores paralelos es atendiendo a las conexiones entre los diferentes elementos que los forman:

- **Arrays sistólicos.** Una forma muy eficiente de conectar los elementos de proceso es realizar estas conexiones de acuerdo con la tarea que van a realizar. La solución no es flexible en absoluto (una vez establecidas las conexiones, los procesadores sólo funcionarán eficientemente con la aplicación para la que fueron adaptados) pero se consiguen soluciones muy optimizadas. El problema fundamental que presentan es que esa optimización sólo ocurre en el proceso de los datos haciendo que la Entrada/Salida sea vital para conseguir unas buenas prestaciones.
- **Redes de Banyan.** Para solucionar la falta de flexibilidad de los arrays sistólicos, se pueden sustituir algunos elementos de proceso por conmutadores programables que permitan cambiar la configuración de la red de procesadores ampliando el abanico de posibilidades a las que es aplicable, esto son las redes de Banyan. Tienen la gran ventaja de que como la conmutación es hardware, no hay pérdida de eficiencia en los cálculos. Algunas aplicaciones de este tipo de redes también solucionan en parte el problema de la E/S dejando los datos fijos y realizando operaciones sucesivamente sobre éstos.
- **Cubos-k.** Los cubos k-binarios conectan 2^k procesadores. Las conexiones se hacen entre los vértices (procesadores) de un cubo de k dimensiones. En la figura aparece el caso $k=3$. Este tipo de conexiones ofrecen muchas rutas de

datos posibles y son muy apropiadas para cierto tipo de problemas como las clasificaciones. Tienen el gran problema de que dado un número demasiado elevado de procesadores la conexión se haga imposible por problemas de cableado, es factible llegar a unos 64000 procesadores. Un ejemplo comercial es la famosa Connection Machine.

- **Cubos de ciclos conectados.** Es un refinamiento de los cubos-k que intenta resolver los problemas de conexión que se plantean. En lugar de colocar un único procesador por vértice, se coloca un anillo de procesadores. El gran potencial de este tipo de conexiones está en problemas que se puedan dividir en varias tareas más pequeñas, que, además, lleven a soluciones que puedan combinarse para dar la solución al problema completo. Esta división de las tareas no es, en muchos casos, obvia y queda a la discreción del programador.

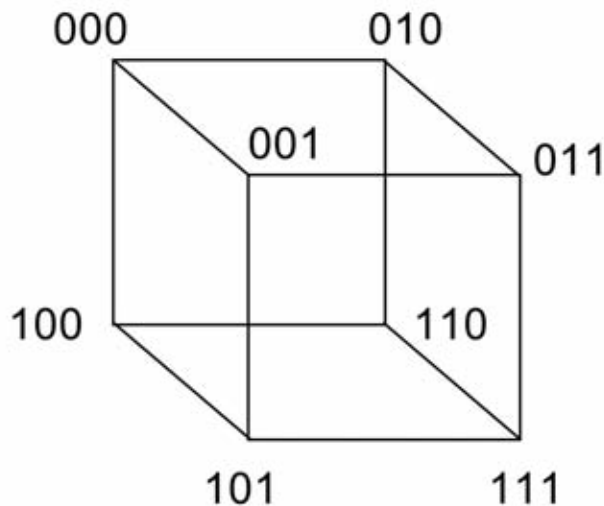


Fig. 10. Esquema de conexión de un cubo K

6.2 Problemas paralelos

El paralelismo es más que una simple curiosidad en el mundo de las arquitecturas de ordenadores. No sólo permite resolver problemas ya conocidos de una forma más eficiente y rápida, sino que también permite abordar problemas totalmente nuevos que ni se podían plantear. Pero las grandes ventajas que ofrecen tienen un costo, a veces muy elevado. Ya hemos mencionado algo sobre este tema al enumerar algunas de las arquitecturas existentes y sus formas de conexión.

La posibilidad de paralelizar los diferentes niveles que se pueden considerar en un ordenador tiene distintas consecuencias según el nivel escogido (al hablar de estos niveles nos referimos a la jerarquía de Bell y Newell. En los niveles más bajos, el paralelismo consigue una mayor velocidad en la ejecución y ciertas ventajas en cuanto a prestaciones del sistema, la duplicación del hardware (para conseguir tolerancia a fallos) o la ejecución segmentada de instrucciones son algunas formas de paralelismo muy utilizadas y con grandes ventajas pero sin que se traduzcan en un aprovechamiento real de lo que es el paralelismo en sí. Otro tipo de paralelismo es el que comentábamos al hablar de los arrays sistólicos, en los que la idea básica es disponer de un número elevado de unidades de cálculo (o de proceso, pero generalmente se utilizan en proceso numérico) que permitan realizar varias operaciones a la vez sobre un conjunto de datos. Según se conecten estas unidades de cálculo se tendrá mayor o menor flexibilidad, tal y como hemos visto anteriormente.

Programación estructurada y paralelismo

Ya quedan lejos los tiempos en que los ordenadores se programaban bit a bit, a través de conmutadores binarios. A la utilización de los lenguajes de ensamblador le siguieron lenguajes como Fortran o PL-1 en los que el grado de abstracción era mucho mayor. Tras éstos aparecieron poco a poco los primeros lenguajes de programación estructurada, orientados a la resolución de problemas por el conocido algoritmo "divide y vencerás". Pero la programación estructurada tiene una serie de problemas que se han ido conociendo con el tiempo. En primer lugar, y al contrario de lo que mantiene la creencia popular, la programación estructurada no facilita el diseño de programas, lo que facilita es su mantenimiento y depuración. Utilizar un lenguaje de este tipo obliga a descomponer el problema de forma coherente y estudiar una jerarquía, a veces muy compleja, de procedimientos y funciones. Hacerlo así facilita la depuración, pues cada procedimiento puede probarse por separado y es más sencillo localizar los posibles fallos. También facilita el mantenimiento pues las modificaciones son generalmente en los procedimientos, no en el programa completo. Pero, al igual que sucede con el paralelismo, y es lo que queremos resaltar aquí, un lenguaje de programación estructurada, al igual que un lenguaje de programación paralelo, traslada gran parte de la complejidad de diseño al programador que ha de resolver el problema de una forma que puede no ser la más adecuada a su estructura. El paralelismo hace que esto sea aún más crítico, pues puede llegar a obligar al programador (macroparalelismo) no sólo a idear una solución paralela sino a controlar la ejecución concurrente y simultánea de varios subprogramas.

Pero el paralelismo realmente interesante es el que presentan los ordenadores que disponen de varios elementos de proceso (varias CPU's), capaces de ejecutar cada uno su propio programa. En este punto es donde comienza la controversia entre las diferentes formas de interpretar el paralelismo. La primera alternativa es elegir entre una arquitectura compuesta de muchos elementos de proceso simples, cada uno de ellos capaz de manipular sólo unos pocos datos (en el límite, procesadores de un bit), o utilizar una arquitectura con bastantes menos procesadores, pero éstos mucho más potentes (procesadores completos como los utilizados en máquinas monoprocesador). Y esta primera elección no es en absoluto trivial pues el conjunto de problemas abordables de forma directa con cada una de las arquitecturas es muy diferente. En cierta forma, el problema es similar al que discutíamos al hablar de los ordenadores RISC y CISC.

Otro punto muy importante dentro del paralelismo es optar por el macroparalelismo o hacerlo por el microparalelismo. Este último propone aprovechar la estructura paralela de algunas construcciones de lenguajes de programación tradicionales, sería el compilador el encargado de traducir el programa, escrito sin tener en cuenta la arquitectura paralela en que se va a ejecutar, a un código preparado para sacar partido de la existencia de varios procesadores. El macroparalelismo, en cambio, se plantea la resolución de problemas a través de algoritmos paralelos, es el programador el que debe estructurar la solución de forma que sea ejecutable en un ordenador paralelo y saque partido de las prestaciones que pueda ofrecer. Son dos enfoques radicalmente diferentes y que se basan en una interpretación distinta de la complejidad inherente a la programación.

Los más pesimistas consideran que no es factible trabajar con ordenadores paralelos, dado el gran número de factores a controlar y su complejidad (las comunicaciones entre los diferentes procesadores y procesos que se ejecutan, ejecución no determinista, efectos laterales, descomposición del problema para poder resolverlo con un algoritmo paralelo, etc.), por lo tanto, el esfuerzo de paralelización ha de hacerse en tiempo de compilación trasladando la complejidad del programador al compilador que se encargaría de optimizar el programa y paralelizarlo (microparalelismo). Esto tiene la ventaja de que no hay que cambiar los hábitos de programación y los programas se ejecutarían mucho más eficientemente. Por ejemplo, un bucle "for i := 1 to 100 do" podría repartir la tarea entre cinco procesadores asignando a cada uno un tramo del recorrido del índice (1-20, 21-40, 41-60, 61-80, 81-100).

Sin embargo, esta solución tampoco está exenta de problemas muy complicados de resolver, entre ellos, la construcción de compiladores capaces de discernir entre estructuras paralelizables y las que no lo son, sin intervención del programador. Por otro lado, existen muchos problemas cuya solución es naturalmente paralela y tener la posibilidad de ejecutarlos de esa manera sería una gran ventaja. Aquí es donde aparecen los partidarios del macroparalelismo, argumentando que es

factible manejar la complejidad que genera un ordenador paralelo a través de las herramientas y los lenguajes de programación adecuados. Aplicaciones como la predicción del tiempo atmosférico, los estudios aerodinámicos, las grandes simulaciones y el proceso de imágenes, son ejemplos de problemas donde el paralelismo se puede utilizar de forma natural, e incluso es la única forma factible de resolverlos. Ciertamente es que, hoy por hoy, existen más ordenadores paralelos que herramientas y lenguajes adecuados para utilizarlos, pero no hay ninguna duda sobre la importancia de este tipo de arquitecturas y el importante papel que van a desempeñar en el futuro de la computación.

7. Resumen

A lo largo de este capítulo hemos tratado algunos aspectos destacados de la tecnología hardware intentando resaltar su importancia en la innovación tecnológica y el papel que juegan en el desarrollo de lo que se ha dado en llamar tecnologías de la información. Por supuesto, nos hemos dejado muchos temas e incluso sobre los que se han tocado habría decir mucho más, ya mencionábamos al principio que el hardware comprende un sinnúmero de cuestiones que sería demasiado complicado y prolijo desarrollar por completo.

De este capítulo conviene destacar dos cosas. Primero, el hecho que hemos intentado resaltar continuamente de que el hardware es el auténtico motor del desarrollo tecnológico y el que propicia las pocas o muchas "revoluciones" que en este campo se dan. Tanto es así, que es frecuente encontrar tecnología en busca de aplicaciones en las que se puedan aprovechar las prestaciones que ofrecen los equipos y esto sucede no sólo con ordenadores sino también con los periféricos (impresoras láser, discos ópticos, etc.) y todo tipo de tecnologías hardware, desde los circuitos integrados a las redes de comunicación para ordenadores. Esto tiene un lado contraproducente que es la enorme velocidad de cambio y evolución de este campo, algo que se traduce en una dificultad enorme para comprenderlo totalmente. A la complejidad inherente al hardware se añade la complejidad de descripción de un objeto polifacético y constantemente en transformación. Por ello es necesario aproximarse al problema con una perspectiva global y para ello nada mejor que las herramientas y estructuras conceptuales que hemos ido desarrollando en capítulos anteriores.

En segundo lugar, un factor importante de la tecnología es la complejidad que ella misma genera. Un ordenador permite manejar complejidad pero al mismo tiempo es un generador de complejidad. El hardware no es una excepción. A medida que surgen equipos más y más potentes con innovaciones que permiten aumentar en muchos órdenes sus prestaciones (y por tanto manejar más complejidad) cada vez es más difícil controlar estos ordenadores. El ejemplo del paralelismo es muy claro en este punto. La enorme ventaja que supone tener varios procesadores trabajando al mismo tiempo se traduce en una enorme complejidad de programación,

independientemente de quién la gestione, el programador o el compilador. Poder integrar circuitos en VLSI implica poder construir ordenadores más potentes y compactos, con menos consumo y más eficientes, pero también implica disponer de las herramientas necesarias para poder diseñar algo tan complejo como un circuito con 100.000 transistores. Esto conduce a una especie de paradoja de la tecnología que puede explicar en parte lo vertiginoso de su avance e incluso determinadas crisis que se producen: la tecnología genera tecnología. Cuando se consigue un equipo con unas prestaciones X el avance que supone respecto a equipos anteriores permite perfeccionar el diseño (y el desarrollo, y las pruebas y las simulaciones, etc.) de nuevos equipos con lo que al tener el equipo X se está en disposición de poder desarrollar un equipo Y aún más potente y así una y otra vez. ¿Cuándo se para ese ciclo?, cuando la tecnología ha avanzado tanto que no hay herramientas software capaces de aprovechar la potencia que ofrecen o no existen aplicaciones en las que esa potencia se pueda utilizar (esto último puede parecer paradójico pero las aplicaciones depende mucho del estado del arte de cada campo concreto, de los intereses que existan en el momento y de otras tecnologías que pueden estar menos avanzadas). Un ejemplo de ello es la tecnología de comunicaciones que siempre va muy por delante de los productos comerciales, ofreciendo anchos de banda varios órdenes de magnitud por encima de lo que se está utilizando.

8. Bibliografía

La dividimos en dos apartados. En el apartado de Notas Bibliográficas se comentarán aquellos trabajos que más profusamente han servido para redactar las páginas anteriores. El apartado de Referencias Bibliográficas contiene todos los trabajos citados.

Notas bibliográficas

Bibliografía sobre clasificaciones de arquitecturas: Los dos artículos que hemos utilizado como referencia para esta parte son dos buenos ejemplos de cómo tratar el problema de las clasificaciones. Además son complementarios ya que el trabajo de Dasgupta pretende completar el de Skillicorn. El artículo de Dasgupta resulta especialmente interesante por el estudio previo que hace de las taxonomías y las características que toda buena clasificación debe reunir. Skillicorn, David B. "A Taxonomy for Computer Architectures", **IEEE Computer**, Noviembre 1988, pp. 46-57; Dasgupta, Subrata, "A hierarchical taxonomic system for computer architectures", **IEEE Computer**, Marzo 1990, pp. 65-74

Bibliografía sobre Ordenadores Personales: Si hay un tema favorito de los estudiosos del mundo de los ordenadores, es sin duda el de los ordenadores personales. Citar incluso una bibliografía básica resumida nos podría llevar páginas enteras. Bástenos aquí con un par de referencias interesantes: un estudio completo

sobre el mundo de los ordenadores personales se puede encontrar en **Computadores Personales**, de F. Sáez Vacas, editado por Fundesco (Madrid, 1987) que, además, tiene la ventaja de estar en español. Una buena referencia a la historia de los pc's y su problemática es el número de Mayo de 1986 del **IEEE Spectrum**, dedicado por completo a los ordenadores personales. Los números de Enero de esta misma revista presentan un análisis del estado del arte de la tecnología, durante los últimos años buena parte del interés se ha centrado en los pc's por lo que puede ser una fuente importante de datos.

Bibliografía sobre arquitecturas RISC: Las referencias utilizadas no entran en detalles demasiado técnicos pero presentan una discusión muy interesante de la aportación de este tipo de arquitecturas y la motivación que conduce a plantearse un problema de este tipo. Gimarc, C.E. y Milutinovic, V.M. "A Survey of RISC Processors and Computers of the Mid-1980s", **IEEE Computer**, Septiembre 1987, pp. 59-69, es un buen artículo para estudiar las características básicas de este tipo de ordenadores y cómo se resuelven los problemas que plantean. En Sáez Vacas, F. **Computadores Personales**, Fundesco 1987, pp. 58-62, se hace un estudio sobre las arquitecturas RISC desde el punto de vista de la complejidad, resaltando sobre todo las implicaciones que conlleva el diseño RISC. Tabak, D. "Which system is RISC?", **IEEE Computer**, Octubre 1986, pp. 85-86, es un artículo breve en el que se discute muy acertadamente cómo se deben clasificar los sistemas RISC y cuáles son las características más sobresalientes de estos ordenadores.

Bibliografía sobre paralelismo: Sobre este tema la bibliografía es abundantísima por lo que nos limitaremos a citar algunas referencias fáciles de localizar y que pueden servir muy bien como introducción a los ordenadores paralelos. Existen dos números del **IEEE Computer** dedicados al paralelismo, uno es el de Agosto de 1986 y otro el de Enero de 1982. En esta misma revista existen otros muchos artículos dedicados a tratar aspectos concretos siempre a un nivel más o menos asequible. Para encontrarlos, los números de Diciembre de todos los años llevan un índice anual en el que la entrada Paralelismo es muy frecuente y con muchos artículos. También hay que citar el **IEEE Spectrum** que en un tono más divulgativo tiene artículos muy interesantes en los que se comenta el estado del arte del paralelismo, en concreto, se puede consultar "A parallel architecture comes of age at last" de P. Wiley, en el número de Junio de 1987, pp. 46-50.

Referencias bibliográficas

Alonso, G. (1989) "**Paralelismo, nuevas proposiciones desde la cibernética**", premio Antonio Fernández Huertas de la Sección Española del IEEE.

Dasgupta, Subrata (1990) "A hierarchical taxonomic system for computer architectures", **IEEE Computer**, Marzo, pp. 65-74.

Engineering Tools (1988), número 1, volumen 1, dedicado a los ordenadores personales y estaciones de trabajo, Febrero.

Fernández G, y Sáez Vacas, F. (1984) **Fundamentos de los ordenadores**, ETSIT, Madrid

Gelernter, D. (1986) "Domesticating parallelism", **IEEE Computer**, Agosto, pp.12-16.

Gimarc, C.E. y Milutinovic, V.M. (1987) "A Survey of RISC Processors and Computers of the Mid-1980s", **IEEE Computer**, Septiembre, pp. 59-69.

Haynes, Siewiorek et al. (1982) "A survey on Highly Parallel Computing", **IEEE Computer**, Enero, pp. 9-22.

LoPiccolo, P.J. (1988) "High-End Workstations", **Engineering Tools**, num. 1, vol. 1, Febrero pp. 80-95.

Matsumura, T. (1983) "Future Microprocessor trends", **Congreso Mundial IFIP**, pp. 213-217.

Murakami et al. (1985) "Research on a Paralell Machine for fifth generation computer systems", **IEEE Computer**, Junio 1986, pp. 76-91.

Pagels, H.R. (1989) **The dreams of reason. The computer and the rise of the sciences of complexity**, Bantam Books, Nueva York.

Sáez Vacas, F. (1983) "Facing informatics via three level complexity views", **X International Congress on Cybernetics**, Namur, Bélgica, pp. 30-40.

Sáez Vacas, F. (1987) **Computadores Personales**, Fundesco, Madrid.

Skillicorn, David B. "A Taxonomy for Computer Architectures", **IEEE Computer**, Noviembre , pp. 46-57.

Tabak, D. (1986) "Which system is RISC?", **IEEE Computer**, Octubre, pp. 85-86.

Wiley, P. (1987) "A parallel architecture comes of age at last", **IEEE Spectrum**, Junio, pp. 46-50.

Capítulo 13: Complejidad y sistemas distribuidos

El fenómeno del microprocesador está dando la vuelta al concepto de informatización que ha estado en boga durante muchos años. Estas pequeñas "pastillas" han puesto una gran potencia de procesamiento en manos de los usuarios, posibilitando capacidades hasta hace poco sólo reservadas a los miniordenadores o a los "mainframes".

Ahora, el reto consiste en coordinar recursos distribuidos, consiguiendo que colaboren eficientemente entre sí, para materializar de forma práctica y abordable su teórico potencial de servicios a los usuarios.

Interconectar ordenadores y lograr su cooperación no es tarea sencilla. Además de los problemas que hay que considerar respecto al hardware y al software, aparece el nuevo y complejo factor adicional de las comunicaciones. A fin de cuentas, la coordinación de recursos es un asunto simultáneo de control y comunicaciones. Por todo ello, los sistemas distribuidos se presentan como un caso claro de emergencia de complejidad.



1. Introducción

Con la proliferación de redes de área local interconectando ordenadores, y su introducción en ámbitos tan variados como la oficina, la factoría de producción o los centros de investigación, se ha hecho posible pensar en la construcción de sistemas informáticos distribuidos. Comienza a hablarse, incluso, de sistemas distribuidos de área amplia, donde los recursos informáticos no solo estarían unidos por redes de área local, sino también por redes públicas de transmisión de datos ¹. Nuestro objetivo en este capítulo no va a ser tan amplio. Nos centraremos en lo que se suele denominar "sistemas distribuidos débilmente acoplados", en el ámbito de redes de área local².

¹Un ejemplo de estos sistemas, aunque aún poco desarrollado y quizás algo sencillo, es el que nos ofrecen los bancos mediante los cajeros automáticos. Cuando nos acercamos a sacar dinero o a consultar nuestro saldo, el terminal nos proporciona la ilusión de que somos el único usuario de un sistema local. Sin embargo, la realidad es muy otra: para que podamos ver los últimos movimientos impresos en un papel, han debido colaborar eficientemente ordenadores situados quizás a varios cientos de kilómetros de distancia.

²En [Coulouris, 1988] se definen los sistemas distribuidos débilmente acoplados como "aquéllos donde los recursos compartidos necesarios para conseguir un servicio informático integrado son aportados por algunos de los ordenadores de la red, y son accedidos por software que corre en todos los ordenadores, usando la red para coordinar su trabajo y para transferir datos entre ellos".

2. Los sistemas distribuidos débilmente acoplados.

Estos sistemas van más allá de la mera compartición de ciertos elementos, como impresoras o servidores de ficheros. Pretenden conseguir que el usuario que interactúe con ellos perciba todos los recursos del sistema como locales, aunque en realidad los obtenga gracias a la colaboración de muchos elementos distribuidos. Esto permitiría una mayor dispersión geográfica de los elementos que colaboran para constituir el sistema. En pocas palabras, la idea es sustituir un gran "main-frame", del que cuelgan recursos centralizados, por una tupida red de pequeños y medianos ordenadores, conectados con una pléyade de recursos distribuidos.

Este cambio, si se realizara así, sin más, plantearía inmensos problemas al usuario. Cada vez que necesitase un servicio, debería saber dónde está, qué características tiene³, cómo se accede a él, prever posibles conflictos con otros usuarios que también lo solicitaran... Y además, todo esto debe ser hecho de una forma eficiente para no malgastar recursos (y aportar un rendimiento al menos similar al de las máquinas centralizadas, a un coste igual o menor). Aún más complicado será el caso en que el servicio no sea dado por un único recurso, sino que necesite de la colaboración de varios (que es el caso más habitual)⁴. Si no dotamos a estos sistemas de algún mecanismo capaz de absorber toda esta complejidad, probablemente nuestro sufrido usuario preferirá, con toda razón, un sistema centralizado. Y los mismos problemas encontrará un programador que intente diseñar software distribuido.

Como breve adelanto, digamos que estos mecanismos existen (o están en fase de investigación y desarrollo). Los usuarios dispondrán de sistemas operativos distribuidos que oculten todos los aspectos "internos" del sistema, ofreciendo una interfaz homogénea (de una forma parecida a como un sistema operativo tradicional lo hace para una máquina aislada). Los programadores pueden ayudarse de herramientas CASE para el diseño distribuido, y lo que es más importante, de modelos teóricos que podrán aplicar para simplificar sus diseños.

³Usualmente se pretende que en los sistemas distribuidos los recursos informáticos no tengan por qué ser de características homogéneas. Únicamente han de soportar protocolos normalizados de acceso a la red, y poco más (al menos, en cuanto a requisitos hardware). La idea es que puedan interconectarse sin problemas incluso equipos de diferentes fabricantes, con diferentes arquitecturas.

⁴Un acto en principio sencillo, como puede ser el caso de un usuario que manda un mensaje a otro a través del correo electrónico, necesita de la colaboración de, al menos, tres procesos (que, en un entorno distribuido, normalmente no estarán en la misma máquina). El primero será el que interactúe con el creador del mensaje (editor de textos, o algo parecido), y se encargue de enviarlo al proceso repartidor. Éste, según la dirección, lo enviará a la máquina correspondiente. Por último, cuando el destinatario quiera leerlo, le hará falta un nuevo proceso que sea capaz de acceder al mensaje, y se lo presente de una forma adecuada.

3. Categorías de complejidad en sistemas distribuidos

Pero antes de entrar con algo más de detalle en estas soluciones, analicemos el problema básico de la complejidad de los sistemas informáticos, desde el punto de vista de la comunicación entre máquinas espacialmente separadas. Podemos distinguir tres categorías o niveles jerárquicos de complejidad relacionados con su diseño y operación:

- a. **Complejidad interna.** Caen aquí los problemas de diseño de software para máquinas aisladas. Se trata con planificadores de procesos, gestores de memoria, etc. Son las tareas típicas de los **sistemas operativos para máquinas aisladas** (esto es, que no se comunican con otros ordenadores)⁵.
- b. **Complejidad de comunicación.** Cuando los ordenadores comienzan a "hablar entre sí", aparece un nuevo nivel de complejidad. El entorno donde aparece este nivel suele estar compuesto por máquinas individuales conectadas mediante enlaces de baja velocidad. Aparecen dos conceptos muy importantes: la asincronía (ya no tenemos un reloj común para todos los procesos) y la posibilidad de fallos locales (puede caerse una máquina, con todos los procesos que en ella corren, mientras el resto del sistema sigue funcionando)⁶. Históricamente, en este nivel hemos tenido (y tenemos) extensiones de los sistemas operativos clásicos, que incluyen mecanismos de comunicación⁷.
- c. **Complejidad de colaboración.** Ahora, los ordenadores no sólo hablan entre sí, sino que han de colaborar de forma que el usuario (ya sea una persona o un proceso informático) perciba el sistema como una entidad única y local (aunque en realidad todos los recursos estén distribuidos). El escenario sobre el que este concepto está comenzando a tomar forma está compuesto por

⁵En [Coulouris, 1988] se citan sistemas operativos centralizados como CTSS, Multics o Unix, que en su configuración más sencilla (sin la parte de comunicaciones), entran perfectamente dentro de esta categoría.

⁶Aparecen también otros problemas, como el control de acceso (necesidad de identificar la máquina con la que hablamos, para permitirle acceso sólo a los recursos a los que tiene derecho), la saturación de enlaces de comunicación y otros problemas de comunicaciones,...

⁷Por ejemplo Unix, que nació como sistema operativo "de máquina aislada" (prácticamente carecía de medios de comunicación con otras máquinas), fue extendido en Berkeley con mecanismos de comunicaciones, que permiten abrir canales de transferencia de información con máquinas remotas (sockets, protocolos TCP/IP, etc.).

máquinas relativamente pequeñas (típicamente estaciones de trabajo y servidores⁸), conectadas mediante una red local de alta velocidad (al menos, 10 Mbit/s). La necesidad de unas buenas primitivas de comunicación, que hagan uso eficiente de la red, se hace imprescindible. Se acentúan las características generales del nivel anterior. Y además emergen otras nuevas, debidas a la necesidad de colaboración. Problemas como la compartición de datos por procesos remotos o la ejecución en paralelo en máquinas distintas no hacen más que aumentar la complejidad. Para actuar en este estrato aparecen los **sistemas operativos distribuidos**⁹. Si bien suelen guardar diferentes grados de compatibilidad y similitud con los de máquina aislada, constituyen esencialmente una nueva forma de entender los sistemas operativos. Será a este nivel al que nos dedicaremos con más amplitud en lo que queda de apartado.

Cada uno de estos niveles implica un salto cualitativo en cuanto al tipo de complejidad que implica. Además, el que en un sistema aparezcan características de complejidad de un nivel dado no excluye que también presente problemas característicos de los niveles inferiores. Por el contrario, puede entenderse que los niveles se estructuran de forma que cada uno engloba a los anteriores¹⁰.

⁸Por ejemplo, en [Mullender, 1990] se describe la arquitectura sobre la que se construye Amoeba (un ejemplo de sistema operativo distribuido). Supone la existencia de estaciones de trabajo (donde trabajarán los usuarios humanos), armarios de procesadores (que proporcionan la mayor parte de la potencia de cálculo), servidores especializados (de ficheros, de bases de datos, de impresión,...) y máquinas de comunicaciones ("gateways", para enlazar con otros sistemas remotos), unidos por una red de área local (normalmente, de tipo Ethernet).

⁹En los últimos años (especialmente a partir de 1980), muchos equipos han desarrollado sistemas operativos distribuidos. Como rápido exponente de estos esfuerzos, se incluye a continuación una lista (incompleta, por supuesto), tomada de [Coulouris, 1988]: Xerox DS (Xerox PARC, 1977), CDCS (Universidad de Cambridge, 1979), Locus (UCLA, 1980), Apollo Domain (Apollo Co., 1980), Newcastle Connection (Universidad de Newcastle, 1980), Grapevine (Xerox PARC, 1981), V-system (Universidad de Stanford, 1982), Argus (MIT, 1983), Amoeba (Universidad Libre de Amsterdam, 1984), Unix+Sun NFS (Sun Micro., 1985), Mach (CMU, 1986), Chorus (Chorus systèmes, 1988),...

¹⁰Por ejemplo, al diseñar un sistema en el ámbito de la complejidad de comunicación, seguiremos teniendo también problemas de complejidad interna.

La **estructuración en tres niveles de las tecnologías de la información** que puede encontrarse en el anexo sobre ofimática concuerda bastante bien con la ofrecida aquí para los sistemas informáticos¹¹. El nivel de proceso de información se relaciona con lo que hemos llamado "complejidad interna", el de comunicación con la "complejidad de comunicación" y el de colaboración con la "complejidad de colaboración". Sin embargo, conviene no olvidar que en el enfoque actual no se ha tenido en cuenta (o casi) la incidencia del usuario humano sobre el sistema, ni la de los elementos que lo componen por sí solos. Por tanto, estamos dentro del ámbito de la complejidad sistémica del modelo de Sáez Vacas.

También podemos situarnos dentro del marco propuesto por Bell y Newell. Si recordamos, había un nivel donde las estructuras relevantes eran los ordenadores, las redes telemáticas, etc. Lo llamábamos (en realidad, lo llaman así Bell y Newell) **nivel PMS**. Ese es el ámbito en el que nos movemos cuando hacemos las distinciones que por el momento vamos exponiendo en este apartado.

4. Algunas soluciones

Una vez descrito el escenario donde nos encontramos, pasemos a discutir brevemente algunas soluciones (a modo de ejemplo) propuestas para los problemas que plantea la complejidad de colaboración en el ámbito de los sistemas distribuidos. No perdamos de vista el hecho de que lo que buscamos (de nuevo desde el **esquema $H \times I \times O = IO$**) son interfaces I que permitan al observador H¹² manejar una imagen del objeto IO más sencilla que el objeto inicial O (en nuestro caso, un sistema distribuido).

4.1 El sistema Amoeba

El **sistema operativo Amoeba** puede ser un buen ejemplo de este tipo de interfaz (entre los otros muchos sistemas operativos distribuidos en fase de desarrollo, o incluso ya comercialmente disponibles). Permite a un usuario (ser humano o proceso) manejar con relativa facilidad "objetos distribuidos"¹³, mediante operaciones del tipo "llamada a procedimiento remoto" (que será descrito brevemente más adelante). Actúa por lo tanto como **filtro de variedad** entre la del sistema

¹¹Y esto no es, por otra parte, extraño. Piense el lector que un sistema ofimático es, entre otras muchas cosas, un caso particular de sistema distribuido.

¹²Como ya hemos adelantado en algún momento, el "observador" no ha de ser forzosamente humano. Aquí se nos presenta un caso en que puede no serlo: cuando, por ejemplo, es un proceso informático, que pretende acceder a algún tipo de servicio distribuido.

¹³En [Mullender, 1990] se describe un objeto Amoeba como "un conjunto de datos sobre los cuales los usuarios autorizados pueden realizar ciertas operaciones, independientemente de la situación física del usuario y del objeto".

distribuido (muy grande) y la del usuario, al limitar los tipos de interacciones posibles entre ambos a las permitidas por los objetos distribuidos disponibles. Para la especificación y uso de estos objetos Amoeba proporciona el AIL¹⁴, que permite realizar estas tareas de forma sencilla. La seguridad viene dada por el uso de "capabilities", objetos abstractos asociados a las operaciones sobre objetos. El sistema da "capabilities" a los usuarios de forma que se asegure la protección del objeto ante usos negligentes o no autorizados. Alrededor de estos dos conceptos, objetos y capabilities (y con la ayuda de unos eficientes protocolos de comunicaciones), se construye el sistema Amoeba. El resultado es un sistema operativo que consigue que para el usuario sean transparentes aspectos como la localización física de los ficheros, la concurrencia de accesos sobre ellos, la replicación de datos o cierto grado de tolerancia frente a fallos¹⁵.

4.2 Las llamadas a procedimiento remoto.

Al principio de este capítulo se habló de la importancia de disponer de herramientas conceptuales para facilitar el diseño de sistemas distribuidos. En lo que queda de él, vamos a describir dos de ellas, para que el lector pueda entender a qué se hacía referencia con esas palabras.

Ya hemos mencionado las **llamadas a procedimiento remoto** (RPC)¹⁶ al hablar de los mecanismos básicos de Amoeba. La idea que sustenta este modelo es extender al ámbito de los sistemas distribuidos la forma "clásica" de solicitar un servicio: la llamada a procedimiento. Se pretende, por tanto, disponer de una interfaz que permita al programador llamar a un procedimiento, pasándole los parámetros de entrada y salida oportunos, sin preocuparse de la localización física donde se ejecutará. El sistema deberá, por lo tanto, descubrir si el procedimiento puede ejecutarse localmente. En caso contrario tendrá que localizar una máquina donde sí se pueda, y comunicarse con ella (pasando y recibiendo los parámetros adecuados). Y todo esto de forma transparente para el usuario. En un caso más general puede ser necesaria la colaboración de varios procesos, localizados posiblemente en máquinas distintas, para ejecutar el procedimiento demandado¹⁷.

¹⁴AIL significa "Amoeba Interface Language" (lenguaje de interfaz con Amoeba).

¹⁵Desde luego, estas breves líneas no pretenden ser una descripción exhaustiva del sistema Amoeba. El lector interesado puede encontrarla en [Mullender, 1990].

¹⁶En inglés, RPC (remote procedure call). Para más información sobre las RPC puede consultarse [Birrell, 1984].

¹⁷Éste es el caso, por ejemplo, de muchos de los servicios del sistema Isis, como el gestor de transacciones o el servicio de noticias (de una forma breve, puede decirse que el sistema Isis es "una extensión al sistema operativo Unix que proporciona una caja de herramientas para facilitar el diseño y la codificación de sistemas distribuidos"). Una información más exhaustiva sobre las características de Isis puede encontrarse en [Birman, 1990].

Desde el punto de vista de la complejidad, las llamadas a procedimiento remoto proporcionan un instrumento que reduce la complejidad que percibe el programador. Una vez construido todo el sistema de llamadas remotas (cosa que, si se hace bien, tendrá una utilidad general), el diseño de un sistema distribuido puede ser tratado como si fuera centralizado. Cuando se necesiten recursos de otras máquinas, llamaremos a los RPCs que gestionen estos recursos, de la misma manera que llamaríamos a un procedimiento local, olvidándonos de todos los problemas de comunicaciones, colaboración entre procesos remotos, etc¹⁸.

Por tanto, constituyen un *I* que permite que el observador perciba un *IO* mucho más sencillo que el *O* original (todo el proceso de comunicaciones y ejecuciones remotas que implementa una RPC). Como modelo conceptual, si el sistema nos "obliga" a realizar todas las interacciones con otras máquinas a través de RPCs, el conjunto virtualmente infinito de posibles formas de realizar las comunicaciones queda reducido a un único elemento: la llamada a procedimiento. Al realizarse todas las interacciones entre máquinas según este modelo, el programador tiene que tener en cuenta menos posibilidades y combinaciones.

4.3 El espacio de tuplas de Linda

El **espacio de tuplas de Linda**¹⁹ proporciona otro mecanismo capaz de facilitar cierto tipo de distribución al programador. Consiste éste en una memoria virtual (organizada como espacio de tuplas), sobre la que se construyen tres operaciones básicas²⁰:

- a. *out*: añade una tupla al espacio.
- b. *in*: lee una tupla del espacio, y la borra.
- c. *read*: lee una tupla sin borrarla.

¹⁸En realidad, no pueden olvidarse totalmente esos problemas. Por un lado, al ir las llamadas por la red, percibiremos un retardo mayor que en el caso de llamadas locales (que se realizan en memoria). Y por otro, el hecho de utilizar recursos remotos da lugar a algunas dificultades si, por ejemplo, alguna máquina se cae mientras ejecutaba una tarea para nosotros. Además, en algún momento ha habido que diseñar los RPCs...

¹⁹Linda S/Net es un sistema operativo desarrollado por AT&T, sobre un hardware especializado en comunicaciones tipo "broadcast" (comunicación de uno a muchos) rápidas. Una descripción detallada del sistema puede encontrarse en [Carriero, 1986].

²⁰La descripción de las operaciones sobre las tuplas está tomada de [Birman, 1988].

En realidad, todo el espacio de tuplas está replicado en las memorias locales de los procesadores. Pero el sistema se encarga de que las puestas al día sean hechas de forma automática (con todas las comunicaciones y transferencias de información necesarias para ello) y transparente para el programador. Se dispone por lo tanto de una memoria que aunque está distribuida y es vista de forma consistente por todos los procesadores, puede ser tratada de una forma sencilla, similar a como se hace con una memoria local.

De nuevo, el programador que usa Linda ve reducida la complejidad, al disponer de un modelo *IO* (el espacio de tuplas) que le permite manejar el objeto *O* (una memoria distribuida y replicada) de una forma muy sencilla. El sistema funciona, por tanto, como una interfaz *I* (según el modelo de Sáez Vacas).

5. Resumen

En este capítulo se ha pretendido aportar una rápida visión de varios aspectos del extenso mundo de los sistemas distribuidos, centrándonos en el ámbito de los débilmente acoplados en red local, por ser los más extendidos hasta el momento. La intención que nos ha movido a ello ha sido la de analizar sus **problemas de complejidad** (y alguna de las aproximaciones empleadas para tratar con ellos). Desde luego, el catálogo no pretende ser exhaustivo, y los problemas tratados deben ser considerados sólo a modo de ejemplo.

Cuando estudiamos estos sistemas, nos enfrentamos a casos de **complejidad sistémica**. Para tener un punto de referencia, se ofrece un sencillo marco donde situar y clasificar los problemas que aparecen, basado en la distinción de **tres categorías diferentes de complejidad**.

En una última parte, se hace un somero repaso de algunos ejemplos de soluciones ya empleadas (o en fase de investigación) en este campo. Concretamente, los casos tratados son el **sistema operativo Amoeba**, las **llamadas a procedimiento remoto** y el **espacio de tuplas de Linda**.

6. Bibliografía

Dividida en dos partes. En primer lugar, Notas Bibliográficas, donde se describen los trabajos consultados más relevantes sobre el tema. Después, Referencias Bibliográficas, donde pueden encontrarse todas las citas utilizadas en el capítulo.

Notas bibliográficas

El libro [Coulouris, 1988], ofrece una amplia perspectiva del campo de los sistemas distribuidos, si bien está orientado específicamente hacia los sistemas operativos. En él pueden encontrarse exposiciones bastante detalladas de la mayoría de los conceptos tratados en el capítulo.

Por otra parte, el ejemplar de mayo de 1990 del IEEE Computer, dedicado por entero a sistemas operativos, incluye muchos artículos de interés para quién desee conocer los esfuerzos de investigación que se están desarrollando hoy en día sobre este particular. Se tratan con especial interés los sistemas operativos distribuidos.

Referencias bibliográficas

Birman, K.P. y Joseph, T.A (1988): "Exploiting replication", **Lecture notes from "Artic 88, an advanced course on operating systems, Trömso, Norway, July 5-14"**.

Birman, K.P. et al. (1990): **The ISIS system manual, version 2.0**, the ISIS Project, Cornell University.

Birrell, A. y Nelson, B. (1984): "Implementing remote procedure call", **ACM Transactions on Computer Systems**, vol.2, núm.1, febrero, pág.39-59.

Carriero, N. y Gelertner, D. (1986): "The Linda S/Net's Linda Kernel", **ACM Transactions on Computer Systems**, vol.4, núm.2, mayo, pág.110-129.

Coulouris, G.F. y Dollimore, J. (1988): **Distributed systems**, Addison-Wesley.

Mullender, S.J. y van Rossum, G. (1990): "Amoeba: a distributed operating system for the 1990s", **IEEE Computer**, vol.23, núm.5, mayo, pág.44-53.

Capítulo 14: Medidas de la complejidad del software

Los programas tienen errores. Desde los pequeños programas comerciales para ordenadores personales, que se venden en los grandes almacenes, hasta los megaprogramas que corren en redes de grandes ordenadores, prácticamente ninguno está a salvo de que de vez en cuando, al cumplirse ciertas condiciones, su funcionamiento sea muy distinto del previsto.

Si consiguiéramos desarrollar código de tal forma que quien lo leyera (o lo escribiera) fuera capaz de comprender exactamente y en todos los casos posibles el comportamiento de ese código, probablemente los errores desaparecerían.

Hoy por hoy nos contentamos con medir la complejidad de los programas, para intentar hacerlos más sencillos, con vistas a que el mantenimiento y corrección de errores sea más fácil de realizar.



1. Introducción

Todo el mundo que haya tenido que escribir alguna vez un programa de tamaño medio (o incluso pequeño) estará de acuerdo en que la producción de software es algo, cuando menos, complicado. Pero para decidir de una forma rigurosa si podemos calificarlo como "complejo", recurriremos a la definición de Sáez Vacas (ver capítulo "Marcos Conceptuales"). Tras la revisión de esta definición, podemos concluir que el software se ajusta a ella al menos en los siguientes aspectos:

- a. "Dificultad de comprensión". Por lo pronto podemos decir sin temor a equivocarnos que el software es normalmente algo difícil de entender. Cuando nos encontramos ante un listado, sólo tras un cuidadoso examen podemos deducir qué es lo que hace el programa correspondiente. Y aún así, esta deducción suele ser aproximada. Lo que verdaderamente sucede en la ejecución sólo podemos saberlo con certeza después, precisamente, de esta ejecución. Si el lector ha tenido que descifrar alguna vez un código escrito por otro programador, o incluso uno suyo, pero desarrollado hace algún tiempo, comprenderá sin duda de qué estamos hablando. La evolución de los lenguajes de programación, por ejemplo, está fuertemente marcada por el intento de conseguir listados más legibles. Por ejemplo Ada, el lenguaje promovido por el Departamento de Defensa de los Estados Unidos, ha sido diseñado buscando de forma especial

el facilitar la comprensión del código. La idea es que un programador diferente del original pueda entender un listado con relativa facilidad, y no le sea difícil hacerse una idea exacta de cómo se comportará el programa correspondiente al ejecutarse. Así suele decirse que Ada es un lenguaje mucho más fácil de leer que de escribir.

- b. "Requiere una gran cantidad de información y tiempo, y el esfuerzo coordinado de personas y maquinaria". En efecto, cualquier proyecto informático de mediana entidad requiere de un equipo de profesionales (analistas, programadores, etc.) que se ocupe de definirlo, desarrollarlo, probarlo, mantenerlo e incluso, quizás, modificarlo, durante toda su vida útil. La gestión del software es un problema hasta la fecha no resuelto en su totalidad, aunque se han propuesto varias técnicas para afrontarlo. Por otra parte, las herramientas CASE (ingeniería software asistida por ordenador) se están desarrollando a gran velocidad, y no por capricho. Para manejar toda la información generada en el proceso de producción de un programa, se hace necesario el contar con estas herramientas. Si no, el problema llega a ser, sencillamente, inabordable. Para hacernos una idea de lo que supone el tiempo en el proceso de desarrollo, basta recordar los repetidos retrasos que a menudo ha sufrido la aparición de populares programas para pc. Son muy conocidos, por ejemplo, los casos del dBase IV y de la nueva versión del Lotus 123. Y es que el tiempo de desarrollo no sólo suele ser largo, sino que también es muy difícil de predecir.
- c. "Efectos positivos y negativos simultáneos". Prácticamente todo el software que se utiliza tiene errores, unos más evidentes y otros menos, unos más peligrosos y otros más benignos, unos conocidos y otros desconocidos (hasta que aparecen, claro, causando normalmente algún perjuicio). Así pues, los programas de que se dispone hacen normalmente lo que se espera de ellos, excepto en el caso de que se encuentren con un error de programación. Los errores no son más que comportamientos no deseados del programa bajo ciertas condiciones que, por alguna razón, pasaron desapercibidas en las etapas de diseño y codificación. Y cuando estas condiciones aparecen, los efectos pueden ser desastrosos. Por ejemplo, a finales de 1988 alguien utilizó defectos en algunos programas de comunicaciones y de correo electrónico ampliamente extendidos para construir un programa que se "reprodujo", sin permiso (y durante cierto tiempo, sin conocimiento) de los usuarios, por gran parte de la red DARPA, en Estados Unidos. Así los programas de comunicaciones que, en principio, producen efectos positivos, produjeron también el efecto negativo de disminuir la seguridad del sistema.
- d. "Comportamiento impredecible". Normalmente, lo más que podemos asegurar de un programa es que ha sido probado bajo muchas condiciones, y que durante esas pruebas no ha presentado comportamientos "extraños". Pero nada más. De hecho, la del software es la única ingeniería que vende sus

productos sin garantía, en el sentido de que es común la venta "tal cual", sin garantizar un determinado comportamiento del sistema informático. Los estudios sobre fiabilidad y verificación constituyen un campo donde la investigación es muy activa, aunque hasta el momento no ha deparado resultados espectaculares (véase el capítulo sobre el **desarrollo del software**).

2. Estudio del software desde el marco HxIxO = IO

Una vez que hemos llegado a la conclusión de que el software es efectivamente algo bastante complejo, podemos analizarlo dentro del marco conceptual **HxIxO -> IO**, propuesto por Sáez Vacas. Para ello, comencemos por identificar cada uno de los elementos de este marco.

De una forma sencilla, podremos identificar el objeto (O) como el propio software, tal y como estamos acostumbrados a entenderlo. Eso es, un programa dado, que una vez ejecutado determina el comportamiento de un ordenador, y que está escrito en algún lenguaje de programación. Consideraremos como objeto el código fuente y no el código objeto porque este primero es con el que se enfrenta el programador (o cualquier otra persona que participa en el diseño, desarrollo o mantenimiento), y del que depende en último extremo el funcionamiento del programa. Estamos asumiendo implícitamente que la herramienta que utilizamos para convertirlo en ejecutable (normalmente un compilador o un intérprete) funciona adecuadamente, sin errores y sin efectos "extraños". Aunque esto no siempre es cierto, no nos planteará demasiados problemas, ya que la herramienta no es más que otro objeto software, al que podríamos aplicar el mismo tratamiento que vamos a aplicar a nuestro programa.

Si queremos enmarcar este objeto dentro del modelo de tres niveles de complejidad, no tendremos problemas en colocarlo en el nivel más bajo, donde nos ocupamos de la complejidad de los elementos aislados (ver capítulo sobre **marcos conceptuales**).

El H de nuestra fórmula puede ser asociado a toda persona que participe en el ciclo de vida de un programa, desde que se especifica su funcionalidad, hasta que se retira de la vida activa. Cada una de ellas tendrá sus intereses y motivaciones, y buscará un aspecto diferente dentro del objeto considerado. Así, al personal encargado del mantenimiento y modificación le interesará que no haya dificultad en la comprensión y sí facilidad para cambiar partes del código. El codificador estará interesado en la capacidad expresiva de las herramientas que use al programar, y en conseguir con el menor esfuerzo posible una codificación eficiente y sin errores. Y así, cada uno de ellos se centrará en unas características diferentes del programa.

Un caso especial aparece cuando identificamos H con el usuario del programa. Aquí O no es propiamente el código fuente, sino el programa en ejecución, y sobre todo su interfaz con la persona que lo usa. En este caso particular (y muy

importante), que no vamos a tratar detalladamente, los aspectos relevantes serán la facilidad de manejo, la convivencialidad de la interacción hombre-máquina, la adecuación de lo que hace el ordenador a las necesidades del usuario, etc.

Por último, nos queda tratar de I. Naturalmente, variará en cada situación concreta, pero en todos los casos será la herramienta que nos permita destacar las características del programa que nos interesan. Por ejemplo, a la hora de mantener un programa, puede ser un buen equipo de ingeniería inversa (reverse engineering), que nos permita trazar la ejecución, las llamadas a módulos, los cambios de valor de las variables, etc. Pero también puede ser de utilidad una herramienta más conceptual, como una métrica de complejidad (de las que hablaremos más adelante). Gracias a su aplicación pueden detectarse, por ejemplo, fragmentos de código especialmente complejos, y por tanto más propensos a que se produzcan errores cuando los modifiquemos. Otras herramientas pueden ser el modelo en cascada de la gestión del software, útil a la hora de planificar el ciclo de vida de un programa (ver capítulo sobre el **desarrollo del software**), el modelo de programación estructurada, que sirve para disminuir la dificultad de seguimiento del flujo de control, o las técnicas de verificación y validación, usadas para analizar programas y determinar si cumplen adecuadamente sus especificaciones.

Así pues, ya tenemos todos los elementos que nos configuran la "imagen del objeto", el IO. Será éste un ente abstracto, un modelo, una especie de proyección del programa real según los instrumentos utilizados, y según los intereses y necesidades del observador. Así, para el programador, constará de un conjunto de sentencias y variables que deben combinarse de cierta forma para producir los resultados apetecidos. Sin embargo, para el que aplica una métrica de complejidad, el programa queda convertido en unos cuantos parámetros que pretenden caracterizarlo. En general, tendremos algo más manejable que el simple listado original, algo que podemos entender mejor.

A esto nos ha ayudado el centrarnos en una parte de las características del problema (según las ideas del observador), y el tratarlas con la ayuda del instrumento correspondiente. La idea, por ejemplo, de ceñirse a la programación estructurada, elimina muchas otras formas de flujo de control posibles a priori. Con esto, el programa resultante es más sencillo de entender por dos razones. Primero, porque sabemos qué estructuras podemos encontrar y cuáles no. Segundo, porque las estructuras elegidas son también más fácilmente comprensibles.

3. Las métricas como instrumentos para abordar la complejidad

Una de las primeras cosas que se nos puede ocurrir para tratar con la complejidad de algo es tratar de medirla. Así podremos establecer una relación de orden, y decir "este objeto es más complejo que este otro".

Además, según dicen, por ejemplo Mills y Dyson, "no se puede controlar lo que no se puede medir" [Mills y Dyson, 1990]. Las métricas de complejidad del software serán el primer eslabón de la cadena que puede llevarnos a controlar esa complejidad. Nos dan una base objetiva para identificar estructuras y técnicas que nos lleven a producir programas de menor (o mayor) complejidad. También nos permiten identificar zonas de un programa especialmente complejas, que sería conveniente rediseñar, y donde probablemente se centren los problemas que aparecerán en la fase de mantenimiento. Normalmente, cuanto más complejo sea un programa, más difícil será su mantenimiento posterior (más propenso a fallos que habrá que arreglar, mayor dificultad de realizar cambios, etc.). Además, al analizar mediante métricas las especificaciones y los primeros documentos del diseño, podemos estimar de una forma más exacta el tiempo de desarrollo [Grady, 1990], o predecir con bastante aproximación la complejidad que tendrá el código [Henry y Selig 1990].

Parece, por lo dicho hasta aquí, que las métricas son muy útiles. Pero ¿qué es lo que realmente miden la métricas?. Según Mills y Dyson [Mills y Dyson 1990], "las métricas [de complejidad del software] son simplemente medidas cuantitativas de ciertas características de un proyecto de desarrollo. Pueden medir alguno de los siguientes objetos:

- a. Productos (como el código o la documentación).
- b. El proceso de desarrollo como tal (aspectos de las actividades del desarrollo).
- c. El dominio del problema (como las telecomunicaciones, los sistemas de tratamiento de información, y el control de procesos).
- d. Las características ambientales (como las personas, las organizaciones y las herramientas)."

Naturalmente, en cada uno de estos casos la utilidad de la medida será diferente, y su significado también. Nosotros nos centraremos en la medida de la complejidad de productos, y más concretamente, de código, por ser uno de los casos más estudiados, y con resultados más contrastados.

La idea de medir la complejidad nos plantea enseguida la pregunta "¿qué tipo de medida podemos y queremos obtener?". Según Baird y Noma [Baird y Noma, 1978], las escalas de medidas pueden dividirse en:

- a. Escalas nominales: La medida simplemente clasifica los elementos medidos (p.e., cuando agrupamos los programas según sean muy difíciles, difíciles, moderadamente difíciles o fáciles de entender).**
- b. Escalas ordinales: Los elementos son ordenados (p.e., A es más difícil de entender que B).**
- c. Escalas de intervalo: La medida nos indica, además de un orden, las "distancias" entre los distintos elementos (p.e., el programa A es 10 unidades de dificultad más difícil que el programa B).**
- d. Escalas proporcionales: Además de determinar la distancia entre elementos, definen un elemento hipotético que tuviera una falta total de las características que se miden, determinando la distancia entre la medida para un elemento cualquiera y para este elemento hipotético (lo que nos permitirá decir, p.e., que el programa A es el doble de difícil de entender que el B).**

Por supuesto, el tipo de escala descrito en el apartado d es el más flexible. Pero también el más complicado de calcular (habría que encontrar un conjunto de características que llevaran asociada una complejidad nula). Lo más normal es disponer de una escala ordinal o, como mucho, una de intervalo (donde nos encontramos con el problema de determinar qué es y qué significado tiene una "unidad de dificultad"). Las métricas que estudiaremos nosotros caen dentro del ámbito de las escalas ordinales.

Desde el modelo $HxIxO \rightarrow IO$, podemos decir que las métricas son un instrumento que, aplicado a alguna parte del proceso de desarrollo de un sistema informático, por alguien interesado en evaluar alguna característica de dicho desarrollo, produce una simplificación del mismo. Y esta simplificación es tan brutal que muchas veces se reduce a un número. Lo que se pretende es que ese número sea lo suficientemente significativo como para que ayude al observador en sus propósitos. Es importante que quede claro también que, aunque como hemos dicho que los objetos sobre los que se puede aplicar una métrica pueden ser muchos, nosotros a partir de este momento nos vamos a centrar únicamente en el estudio de problemas donde O es un programa.

4. Tipos de métricas

Antes de tratar de describir ninguna métrica es precisa una reflexión sobre qué elementos pueden contribuir a aumentar la complejidad de un programa. Dicho de una forma sencilla, una vez identificados estos elementos, bastará con cuantificarlos de alguna forma razonable para obtener esa métrica. Su validez y utilidad habrá de demostrarse a posteriori mediante estudios con casos reales. Estos estudios suelen consistir en la medida, en casos reales, de alguna de las características que dependen de la complejidad (por ejemplo, los recursos dedicados a mantenimiento), y compararla con los resultados de aplicar la métrica, intentando establecer alguna correlación.

Según Basili y Turner [Basili y Turner, 1975], son cuatro los elementos que influyen en el mantenimiento del software:

- a. El tamaño del programa. Los programas grandes requieren más esfuerzo que los pequeños a la hora de efectuar modificaciones, corregir errores, etc.
- b. Las estructuras de datos. Si la organización de los datos es complicada y difícil de entender, el mantenimiento será también más complicado.
- c. El flujo de datos. Por ejemplo, si hay muchas variables comunes a todo el código, que cambian de valor en muchos puntos distintos y que determinan fuertemente el comportamiento del programa, tendremos mucha dificultad en asegurar que un cambio no produce efectos colaterales indeseados.
- d. El flujo de control. El caso de las sentencias tipo GOTO es muy claro. Su utilización produce flujos muy difíciles de seguir y de entender completamente. Cuando hacemos una pequeña modificación, es fácil que cambie el flujo de control de una forma indeseada.

Apoyándonos en estos cuatro aspectos que trataremos de medir en un programa, podemos clasificar las métricas según cual de ellos traten de cuantificar. Así tendremos:

- a. Métricas del tamaño del programa.
- b. Métricas de la estructura y el flujo de datos.
- c. Métricas del flujo de control.
- d. Métricas mixtas. Intentan medir simultáneamente varias características de las anteriores.

5. Métricas del tamaño del programa

Parece claro que los programas muy grandes son complejos, aunque sólo sea por la gran cantidad de información que hay que considerar para poderlos entender. Según esta apreciación, una primera medida de la complejidad del programa vendrá dada por su tamaño. El problema está en definir qué es el tamaño de un programa. O dicho de otra forma, qué es lo que hace que un programa sea considerado como grande o pequeño. Según sea esta definición, tenemos las siguientes métricas:

- a. Número de líneas.
- b. Métricas de Halstead.

Ambas son fáciles de calcular y ampliamente utilizadas. La primera destaca por su sencillez. El segundo apartado incluye en realidad varias métricas, cada una de ellas centrada en diferentes aspectos de la complejidad software.

5.1 Número de líneas

Podemos suponer que, en igualdad de otros factores, es más complejo el programa más grande. La suposición implícita que sustenta esta afirmación es que un objeto es más complejo cuando manejamos más información para describirlo. Y el tamaño del programa está en relación directa con la cantidad de información que lo describe (el código de un programa es precisamente información detallada de cómo se ejecuta).

Contar la cantidad de líneas de código de un programa es una forma sencilla de medir su tamaño. El problema principal de esta sencilla métrica estriba en decidir qué consideramos como línea. Según sea la decisión, estaremos midiendo cosas diferentes:

- a. En primer lugar, el mismo programa dará diferente medida según el lenguaje en que sea codificado (aunque esto tal vez esté en relación con una mayor o menor complejidad "intrínseca" del lenguaje). Así, al comparar tamaños de programas escritos en diferentes lenguajes, no sólo estamos comparando la complejidad de los programas, sino también la de los lenguajes utilizados. Por eso, para quedarnos con la componente debida al propio programa, se hace necesario realizar comparaciones para el mismo lenguaje.
- b. En segundo, tenemos las líneas de comentario. Aunque puedan añadir más complejidad al aumentar la cantidad de información que hay que asimilar, también simplifican la comprensión del código. Por tanto, si las contamos, nos estaremos centrando más en la cantidad de información total. Si no las contamos, nos centraremos sin embargo en la aportada sólo por el código.

- c. Por último, están las líneas de declaración de datos. Quizás no deberían considerarse de la misma manera que las líneas de código, ya que la diferencia conceptual entre ambas es muy clara.

Según el criterio que sigamos, tendremos una métrica distinta. Posiblemente, la más usada sea la cuenta pura y simple de todas las líneas del programa, por ser la más sencilla.

5.2 Métricas de Halstead

Una forma más precisa de medir el tamaño de un programa fue propuesta por Halstead, como parte de su **ciencia del software** [Halstead, 1977]. Para ello, considera que el código está formado por unas unidades que llama operadores y operandos, parecidos a los tokens que un compilador puede distinguir en ese código [Henry y Selig, 1990]. Y estos operadores y operandos no contribuyen siempre de igual forma a la complejidad. Es necesario considerar, además del número total de elementos (operadores y operandos), el número de éstos que son diferentes (esto es, el vocabulario del programa). Investigando las relaciones entre estas cuentas, obtendremos unos cuantos parámetros que intentarán medir diferentes aspectos de la complejidad del programa.

A continuación pasamos a estudiar con algo de detalle esta métrica. Para ello utilizaremos la siguiente notación:

n_1 : número de operadores diferentes.

n_2 : número de operandos diferentes.

N_1 : número total de operadores.

N_2 : número total de operandos.

n : vocabulario de un programa ($n=n_1+n_2$).

N : longitud total del programa ($N=N_1+N_2$).

```

a1.
BEGIN
  FOR I:=2 TO N DO
    FOR J:=1 TO I DO
      IF (X(I) < X(J)) THEN
        BEGIN
          AUX := X(I);
          X(I) := X(J);
          X(J) := AUX;
        END;
      END;
    END;
  END;
a2.
SORT(X,N)
    
```

Fig.1 Implementación real (a1) y potencial(a2), en PASCAL, de un programa de ordenación por el método de la burbuja

```

b1.
{
  for (i=2;i<=n;i++)
    for (j=1;j<=i;j++)
      if (x[i] < x[j])
        {
          aux = x[i];
          x[i] = x[j];
          x[j] = aux;
        }
}
b2.
sort(x,n)
    
```

Fig. 2 Implementación real (b1) y potencial(b2), en C, de un programa de ordenación por el método de la burbuja

Según Halstead, se puede calcular de forma bastante aproximada el valor de N (longitud total) mediante la fórmula

$$N^* = n_1 \log_2 n_1 + n_2 \log_2 n_2$$

BEGIN...END	2	
FOR..TO..DO	2	
:=	5	
IF...THEN	1	
;	5	
(...)	7	
<	1	
n₁=7		N₁=23

Fig. 3 Operadores en a1

{...}	2
for(;;)	2
=	5
if	1
;	3
(...)	1
<	1
<=	2
++	2
[..]	4
n₁=10	
	N₁=23

Fig. 4 Operadores en b1

SORT(...)	1
$n_1^*=1$	$N_1^*=1$

Fig. 5 Operadores en b1

sort(...)	1
$n_1^*=1$	$N_1^*=1$

Fig. 6 Operadores en b2

N proporciona una medida similar a la del apartado anterior (líneas de código), y que está, como ella, en relación directa con la cantidad de información total del programa. Como puede observarse, la estimación de N puede hacerse a partir únicamente del vocabulario empleado. Por lo tanto la complejidad debida al tamaño del programa, depende en última instancia sólo de ese vocabulario.

Además, se define el volumen de un programa como

$$V = N \log_2 n$$

Y el volumen potencial como

$$V^* = N^* \log_2 n^*$$

donde n^* es el tamaño del vocabulario potencial. El vocabulario potencial es algo así como el vocabulario mínimo que podría usar una codificación que hiciera lo mismo que la estudiada. Se corresponde normalmente con el de la llamada a una función que haga la tarea deseada. Se usa como referencia para obtener parámetros normalizados y por tanto más comparables.

I	5
N	1
J	4
X	6
AUX	2
$n_2=5$	$N_2=18$

Fig. 7 Operandos en a1

i	7
n	1
j	6
x	6
aux	2
$n_2=5$	$N_2=22$

Fig. 8 Operandos en b1

N	1
X	1
$n_2^*=2$	$N_2^*=2$

Fig. 9 Operandos en a2

n	1
x	1
$n_2^*=2$	$N_2^*=2$

Fig. 10 Operandos en b2

El volumen pretende ser una medida más precisa de la dificultad en comprender un programa, al tener en cuenta no sólo su "longitud" (N), sino también su vocabulario. Y es que parece lógico que a igual tamaño sea más sencillo un programa con poco vocabulario que otro con mucho.

El nivel de un programa da una idea del nivel de detalle con que ha sido codificado. Se entiende que cuanto más código se usa para una función dada, de más bajo nivel es. En el límite, las llamadas a función tienen el nivel más alto, ya que su volumen real coincide con el potencial. La definición concreta del nivel del programa es

$$L=V^*/V$$

que puede aproximarse por

$$L=(2/n_1) (n_2/N_2)$$

con una correlación de 0,90. Como parece natural, un programa dado es de más bajo nivel cuanto mayor es su volumen (respecto de su volumen potencial).

Con el volumen y el nivel del programa, se calcula lo que llamaremos "inteligencia contenida en el programa":

$$I=L.V$$

Según Halstead, este valor se correlaciona bastante bien con el tiempo total de programación y depuración. Es por tanto una métrica que sirve para estimar la complejidad del programa desde estos dos puntos de vista. Además, parece permanecer bastante invariante cuando cambiamos de lenguaje de programación. Será por tanto una métrica que no mide los aspectos relacionados con las ventajas o desventajas de los distintos lenguajes, sino los del código empleado.

Otro valor interesante de conocer para un programa, es el llamado esfuerzo:

$$E=V/L$$

El esfuerzo necesario para producir una porción de software está relacionado sobre todo con la dificultad de entenderlo e implementarlo. Por eso E puede ser usado como una medida de la claridad de un programa.

Así pues, la "ciencia del software" de Halstead nos ofrece varias métricas que se centran en distintos aspectos de la complejidad del software. Además, permiten estimar de una forma bastante directa tiempos de desarrollo y de prueba (mediante el parámetro I), y dificultad de comprensión (mediante el parámetro E).

	Prog. a	Prog. b
Operadores diferentes n₁ :	7	10
Operandos diferentes n₂ :	5	5
Total de operadores N₁ :	23	23
Total de operandos N₂ :	18	22
Operadores diferentes pot. n₁* :	1	1
Operandos diferentes pot. n₂* :	2	2
Total de operadores pot. N₁* :	1	1
Total de operandos pot. N₂* :	2	2
Vocabulario n (n₁+n₂):	12	15
Longitud observada N (N₁+N₂):	41	45
Long. estimada N* (n₁log₂n₁+n₂log₂n₂):	31,2	44,8
Volumen V (Nlog₂n):	146,9	175,8
Volumen potencial V* (N*log₂n*):	4,75	4,75
Nivel del programa L (V*/V):	0,032	0,027
Contenido de inteligencia I (L·V):	4,70	4,74
Esfuerzo E (V/L):	4590	6511

Fig. 11 Parámetros de Halstead para los programas a y b, de ordenación según el método de la burbuja (código en figuras 1 y 2)

6. Estructura y flujo de datos

Un factor importante en la complejidad del software viene dado por la forma cómo se manejan los datos dentro del programa. A igualdad de otros factores, será más complejo aquél en el que los datos se manejen de una forma más complicada. Según los criterios que tomemos en cuenta para medir esta complicación, tendremos diferentes métricas:

- a. Intervalo entre referencias a datos. Tiene en cuenta que al programador le es más difícil seguir el flujo de datos cuando las referencias a éstos están muy separadas.
- b. Par de uso segmento-global. Cuanto más se usen variables globales, más posibilidades hay de cometer errores, y más complicado es mantener claro lo que pasa con los datos.
- c. Medida Q de Chapin. Se supone que la complejidad que introduce el manejo de datos no es igual para todos los tipos de variables.

La mayoría de estas técnicas tienen problemas a la hora de su utilización práctica, al poderse aplicar sólo a determinados tipos de software.

6.1. Intervalo entre referencias a datos

Esta técnica se apoya en la idea de que cuanto más dispersas por el código estén las referencias a una variable, más difícil de entender será el comportamiento de esa variable. Con una gran dispersión, el programador tendrá que tener en la cabeza los posibles cambios de valor en zonas del listado muy separadas entre sí. Sin embargo, con dispersión pequeña, podrá centrarse más en el código adyacente al que está estudiando. Por tanto, cuando las referencias a variables se separan, el programa será más difícil de mantener y los efectos colaterales indeseados, difíciles de evitar. En general, el software tendrá una mayor complejidad.

Antes de ver cómo se calcula el valor de la métrica, definiremos el concepto de intervalo entre referencias. Será el número de sentencias que hay en el listado de un programa entre dos referencias inmediatas al mismo identificador de variable (ver figura 12). Una vez definido este concepto, el cálculo de la métrica es inmediato. Se reduce a contar la cantidad de intervalos entre referencias que sean mayores que un cierto intervalo.

Esta medida podrá dar una idea de lo difícil que es entender el comportamiento de los datos. Por ejemplo, un programa con un 18% de intervalos mayores de 100, será más complejo que uno que tenga sólo un 7% de intervalos de ese tipo (a igualdad de otras características).

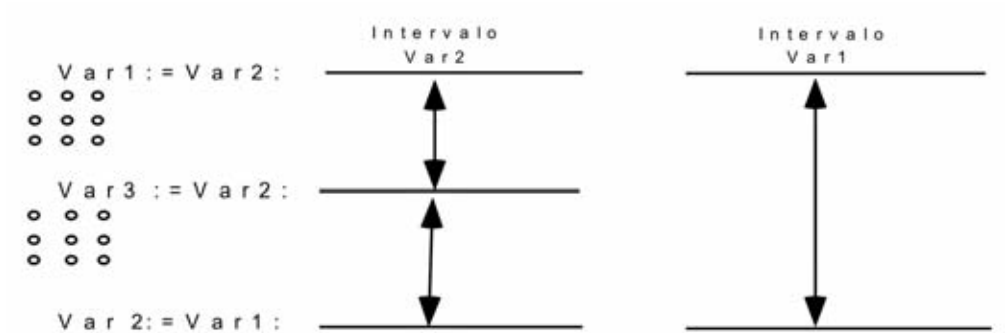


Fig. 12 Intervalos entre referencias a datos

6.2. Par de uso segmento-global

Esta medida intenta dar una idea de la cantidad de veces que un segmento arbitrario de programa accede a alguna variable global. Se supone que si se hacen muchos accesos de este tipo, es fácil que el programador cometa errores, y se produzcan efectos colaterales indeseados en un segmento cuando cambia el valor de una variable global en otro. Estos errores son especialmente posibles en la fase de mantenimiento y modificación del programa.

Dados un segmento de código p y una variable global r , se define el par de uso segmento-global (p,r) como un indicador de que el segmento p usa la variable r . Dicho de otra forma, r es accedida dentro de p . Se define el par de uso real (AUP en inglés) como el número de veces que un módulo utiliza una variable global. Y el par de uso potencial (en inglés, PUP), como el número de veces que un módulo podría acceder a una variable global (se entiende que un módulo p podría acceder a una variable r si p se encuentra en el ámbito de r).

Por último, se define el porcentaje relativo de uso real (RUP) como $RUP=AUP/PUP$. Gracias a esta fórmula obtenemos una medida aproximada de cuánto se usan datos globales dentro de un segmento arbitrario de código.

Sea un programa con cuatro variables globales: Var1, Var2, Var3 y Var4. El programa está compuesto por tres procedimientos: Proc1, Proc2 y Proc3. Si el ámbito de las cuatro variables incluye a los tres procedimientos, tendremos doce pares de uso potenciales:

(Proc1,Var1) (Proc1,Var2) (Proc1,Var3) (Proc1,Var4)
(Proc2,Var1) (Proc2,Var2) (Proc2,Var3) (Proc2,Var4)
(Proc3,Var1) (Proc3,Var2) (Proc3,Var3) (Proc3,Var4)

Por tanto, el PUP será 12. Supongamos que en el programa que estamos analizando Proc1 llama a tres variables, Proc2 a las cuatro y Proc3 sólo a una. Así, $AUP=3+4+1=8$. Si calculamos ahora el RUP nos queda $RUP=AUP/PUP=8/12$. Tenemos, pues, que $RUP=2/3$.

Fig. 13 Ejemplo de cálculo del PUP

Como se desprende de la definición, esta métrica sólo puede usarse cuando el código está dividido en segmentos y hay variables globales, lo que no siempre es cierto.

6.3 Medida Q de Chapin

Ahora los datos son tratados según su uso dentro de cada segmento de código. Para ello son divididos en cuatro categorías, como puede verse en [Chapin, 1979]:

- a. Datos de tipo P: Son los datos de entrada necesarios para que el segmento de código considerado produzca una salida.
- b. Datos de tipo M: Datos que se crean o cuyo valor cambia dentro del segmento.
- c. Datos de tipo C: Datos que se usan para ejercer un papel de control dentro del segmento.
- d. Datos de tipo T: Los que pasan a través del segmento sin experimentar cambios.

Naturalmente, un mismo dato puede representar diferentes papeles dentro del segmento de código. En este caso, se contará una vez en cada clase a la que pueda pertenecer.

Chapin considera que no todos los tipos de datos descritos contribuyen con igual cantidad a la complejidad global del código que estamos estudiando. Los datos de control (tipo C) serán los que más complejidad producen (ya que deciden cuál será el curso de la ejecución, y qué módulos serán llamados). Luego vendrían los datos

de tipo M, y los P (que suelen ser usados para modificar los M). Por último, los datos tipo T casi no contribuyen a la complejidad, ya que simplemente "pasan por el módulo". A partir de estas consideraciones, asignaremos un factor de ponderación diferente a cada uno de los tipos de datos: 3 para el tipo C, 2 para el M, 1 para el P y 0,5 para el T. Un poco más adelante veremos cómo se utilizan estas ponderaciones.

La métrica de Chapin intenta medir la complejidad de la comunicación de datos entre los módulos. Por ello sólo considera los datos que recibe o que envía cada módulo. El algoritmo que usa para calcular su medida de complejidad de un programa (Q) está recogido en la figura 14.

```

PROCEDURE AnalizaTexto;
VAR
  FinDatos: BOOLEAN;
  Palabra: TPALABRA; (* Tipo que sirve para almacenar cadenas de caracteres. Asociado
con él está la constante PAL_VACIA, palabra vacía. *)
BEGIN
  FinDatos := FALSE;
  WHILE (NOT FinDatos) DO
    BEGIN
      LeePalabra (OUT Palabra, OUT FinDatos);
      IF (NOT FinDatos) THEN
        ProcesaPalabra (IN Palabra);
      END;
    END;
  END;

PROCEDURE LeePalabra (OUT Palabra, OUT FinDatos);
VAR
  FinPalabra: BOOLEAN;
  Caracter: CHAR;
 Codigo: INTEGER;
BEGIN
  Palabra := PAL_VACIA;
  FinPalabra := FALSE;
  WHILE ((NOT FinPalabra) AND (NOT FinDatos)) DO
    BEGIN
      LeeCodigo (OUT Codigo, OUT FinDatos);
      ConvCodigo (IN Codigo, OUT Caracter, OUT FinPalabra);
      IF (NOT FinPalabra) THEN
        Palabra := Palabra+Caracter;
      END;
    END;
  END;
END;

```

Fig. 14. Parte de la codificación del programa de ejemplo AnalizaTexto, que utilizaremos para ilustrar el cálculo de la Q de Chapin

```

PROCEDURE LeeCodigo (OUT Codigo, OUT FinDatos);
(* Obtiene un dato, que devuelve en Codigo y detecta si se ha producido el fin de los datos. *)

PROCEDURE ConvCodigo (IN Codigo, OUT Caracter, OUT FinPalabra);
(* Convierte el código ASCII que recibe en el carácter correspondiente, y además detecta si es un carácter de fin de palabra. *)

PROCEDURE ProcesaPalabra (IN Palabra);
(* Procesa la palabra que recibe. *)
    
```

Fig. 14 (cont.) Parte de la codificación del programa de ejemplo AnalizaTexto, que utilizaremos para ilustrar el cálculo de la Q de Chapin

a. Para cada módulo, crearemos una tabla donde aparezcan todas las variables que usa para comunicarse con los demás, junto con sus tipos. Por un lado pondremos las que recibe de otros (y que sean de tipos C, P o T). Por otro, las que otros reciben de él (de tipos M o T). Por supuesto, una variable puede estar en los dos lados de la tabla. En esta etapa puede ser de gran ayuda un esquema como el de la figura 15.

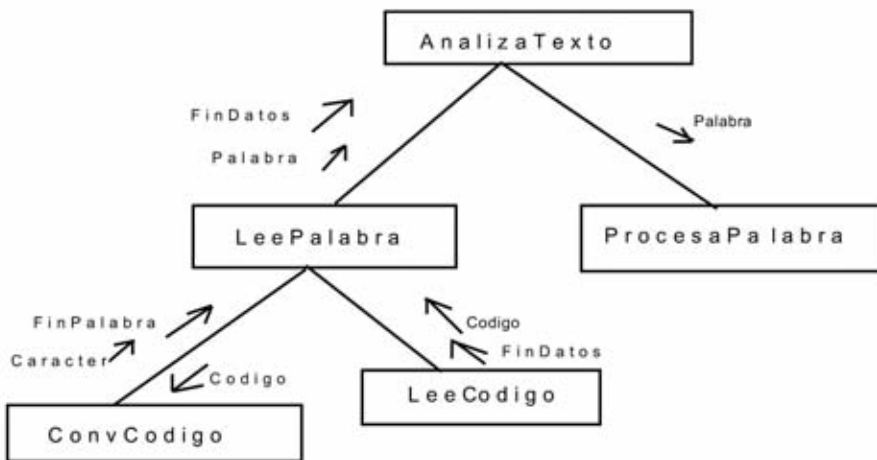


Fig. 15 Estructura de llamadas y comunicación entre los módulos del programa ejemplo Analiza Texto

- b. También para cada módulo, calcularemos el número de variables de cada tipo, y las ponderaremos según los factores descritos unas líneas más arriba. A partir de aquí hallaremos el parámetro W' del módulo como la suma de estos números ponderados.

$$W' = 3 n_C + 2 n_M + n_P + 0,5 n_T$$

Este parámetro está asociado a la complejidad que aportan las variables por su cantidad y tipo.

- c. Otro parámetro que calcularemos para cada módulo es R (factor de repetición), que tiene en cuenta el incremento de complejidad debido a la comunicación de datos entre segmentos que se llaman iterativamente (dentro de un bucle). Para calcular su valor, comenzamos por determinar qué módulos contienen bucles con condiciones de control que incluyen la llamada de más de un módulo. Para cada dato de tipo C que haya en cada uno de los módulos hallados, cuyo valor venga de fuera del cuerpo del bucle, sumaremos 2 al factor E (salida de iteración), que ha sido previamente inicializado a 0 para cada segmento. Si el dato de tipo C es creado o modificado en otro segmento que no sea en el que está la condición de control, pero que está todavía dentro del ámbito de la iteración, sumaremos 1 a E . Y una vez calculado E , el parámetro R vendrá dado por:

$$R = \left(\frac{1}{3} E \right)^2 + 1$$

Naturalmente, si el segmento no hace un test de salida de iteración, E vale 0 y $R=1$.

d. El índice de complejidad de cada módulo (Q), se calcula a partir de los valores de R y W' (que por tanto, pueden considerarse como "medidas intermedias"):

MODULO	VARIABLE	FUENTE	TIPO	DESTINO	TIPO
AnalizaTexto	Palabra	LeePalabra	T	ProcesaPalabra	T
	FinDatos	LeePalabra	C		
LeePalabra	Carácter	ConvCodigo	P		
	Código	LeeCodigo	T	ConvCodigo	T
	FinDatos	LeeCodigo	CT	AnalizaTexto	T
	FinPalabra	ConvCodigo	C		
	Palabra			AnalizaTexto	M
LeeCodigo	Codigo			LeePalabra	M
	FinDatos			LeePalabra	M
ConvCodigo	Codigo	LeePalabra	P		
	Caracter			LeePalabra	M
	FinPalabra			LeePalabra	M
ProcesaPalabra	Palabra	AnalizaTexto	P		

Fig. 16. Tabla de variables de comunicación entre los módulos de Analiza Texto, para facilitar el cálculo de los parámetros de Chapin

e. La complejidad del programa entero se calcula como la media aritmética de las complejidades individuales de los segmentos.

	C	M	P	T	W'	E	R	Q
AnalizaTexto	1	0	0	2	4	1	1,11	2,10
LeePalabra	2	1	1	4	11	2	1,44	3,97
LeeCodigo	0	2	0	0	4	0	1	2
ConvCodigo	0	2	1	0	5	0	1	2,23
ProcesaPalabra	0	0	1	0	1	0	1	1
La Q total del programa será:								
$Q=(2,10+3,97+2+2,23+1)/5=2,26$								

Fig. 17 Tabla que muestra el valor de los parámetros de Chapin calculados para el programa Analiza Texto

De la descripción de la métrica de Chapin se desprende la necesidad de que el software a que se aplica esté estructurado como segmentos de código que se comunican entre ellos, lo que impone una seria limitación a su uso. Además, a pesar de su relativa dificultad de cálculo, no mide importantes aspectos de la complejidad que aportan los datos (como por ejemplo, la debida a la separación entre referencias a variables).

7. Estructuras de control del programa

La posibilidad de que el flujo de ejecución de un programa siga diversos caminos según se cumplan o no ciertas condiciones aumenta de una forma decisiva nuestra dificultad para entender lo que hace el programa en cada una de las situaciones que se pueden dar. Por ejemplo, un pequeño programa con 20 estructuras IF..THEN..ELSE anidadas podrá ejecutarse siguiendo 2^{20} caminos distintos. Y es previsible que tras cada uno de ellos los resultados de la computación sean diferentes, lo que hace que el programador deba tenerlos todos en cuenta (al menos en sus líneas generales).

Normalmente, la complejidad debida al flujo de control se mide contando las transferencias de control que pueden darse en el código (teniendo en cuenta también la longitud total del programa), o estudiando sus interrelaciones.

La explosión combinatoria de caminos posibles en cuanto tenemos unas pocas sentencias de bifurcación, hace necesario el uso de alguna herramienta para poder manejar esa complejidad. Una de estas herramientas, que permite representar los posibles flujos de ejecución dentro de un código, es el grafo orientado. En él, los nodos son bloques donde la ejecución es totalmente secuencial. Sólo disponen de una entrada y una salida, y no hay dentro de ellos saltos en el flujo de ejecución. Los arcos sirven para indicar las posibles transferencias de control de unos nodos a otros. Un ejemplo de grafo, que podría corresponder a un programa real es el de la figura 18.

Como ejemplos significativos de métricas que tratan con la complejidad del flujo de control, describiremos el número ciclomático de McCabe y la extensión que Myers hace del mismo.

7.1. Número ciclomático

McCabe [McCabe, 1976] propone una medida de la complejidad de un programa, basada en su grafo de control, que ha sido ampliamente aceptada. Su éxito probablemente se deba, entre otras causas, a la gran facilidad con que se calcula, a que su significado es intuitivamente sencillo de asimilar, y a que estudios sobre programas reales avalan su relación con el tiempo de desarrollo, la dificultad de mantenimiento, etc.

Esta medida no es otra que el número ciclomático, que se define para un grafo dado. Para calcularlo, supongamos que tenemos un grafo correspondiente al flujo de control de un programa, con a arcos, n nodos y c componentes conectados (normalmente, c valdrá 1). La complejidad ciclomática de ese programa vendrá dada por la siguiente fórmula:

$$V(G)=a-n+2 c$$

El número ciclomático puede entenderse como el número mínimo de caminos necesario para, mediante combinaciones, construir cualquier otro camino presente en el grafo. Utilizamos el término camino en el sentido usual, como una sucesión de nodos que puede recorrerse siguiendo arcos presentes en el grafo.

Por ejemplo, el grafo de la figura 18, donde $a=12$ y $n=10$, tiene una complejidad ciclomática que puede calcularse como:

$$V(G)=12-10+2.1=4$$

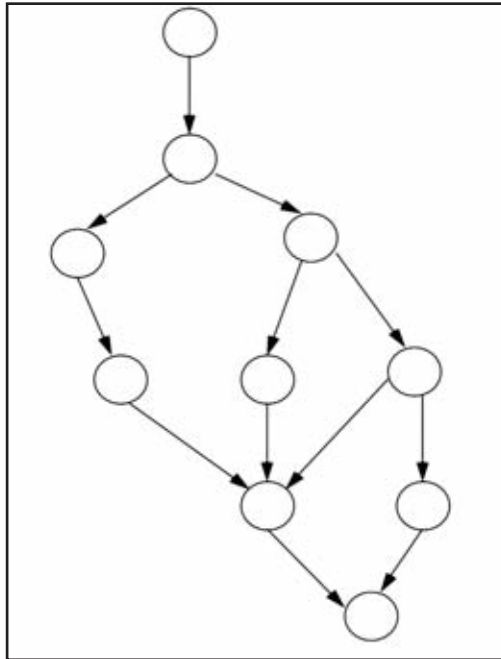


Fig. 18 Ejemplo de grafo correspondiente a un fragmento de programa

Podemos definir un grafo fuertemente conectado como aquél en el cual desde todos y cada uno de sus nodos puede encontrarse un camino que nos lleve a uno cualquiera de los demás. Así, el grafo de la figura 18 no estará fuertemente conectado, ya que desde el nodo inferior, por ejemplo, no puede alcanzarse el superior. Pues bien, basta con añadir un nuevo arco desde este nodo inferior al superior para que sea un grafo fuertemente conectado (ver figura 19). Al calcular el número ciclomático del nuevo grafo, observamos que es

$$V(G)=13-10+2.1=5$$

esto es, uno más que el de la figura 18. Por eso parece lógico añadir uno al número ciclomático calculado para un programa, si lo hemos representado como un grafo no fuertemente conectado.

Dado un programa, codificado en cualquier lenguaje, puede demostrarse que su número ciclomático (suponiendo el grafo fuertemente conectado, en el sentido que hemos visto hace un momento) coincide con la expresión

$$V(G)=N+1$$

donde N es el número total de sentencias de decisión del programa. Además, la cuenta de estas sentencias se hace de la siguiente forma: si es de tipo IF..THEN..ELSE, se cuenta como 1; si es de tipo CASE, con n alternativas, se cuenta como $n-1$. Es sencillo comprobar que esta forma de contar se corresponde exactamente con el número de arcos y nodos implicados en la representación gráfica de estas estructuras.

Esta propiedad del número ciclomático es la que hace que su cálculo sea tan sencillo. Basta con construir un analizador léxico que vaya contando las ocurrencias de las sentencias tipo IF..THEN..ELSE y CASE, con las salvedades explicadas, y la obtención del valor de esta métrica es inmediata.

7.2. Extensión de Myers al número ciclomático

Parece claro que son más complejas las sentencias con condiciones compuestas que las que solo dependen de una condición. Por ejemplo, la condición

```
IF Llueve
  THEN Paraguas
  ELSE NoParaguas
```

es más sencilla que esta otra,

```
IF ((Llueve AND NoCoche) OR (NOT Llueve AND Nubes))
  THEN Paraguas
  ELSE NoParaguas
```

(donde Llueve, NoCoche y Nubes son condiciones y Paraguas y NoParaguas son fragmentos de código).

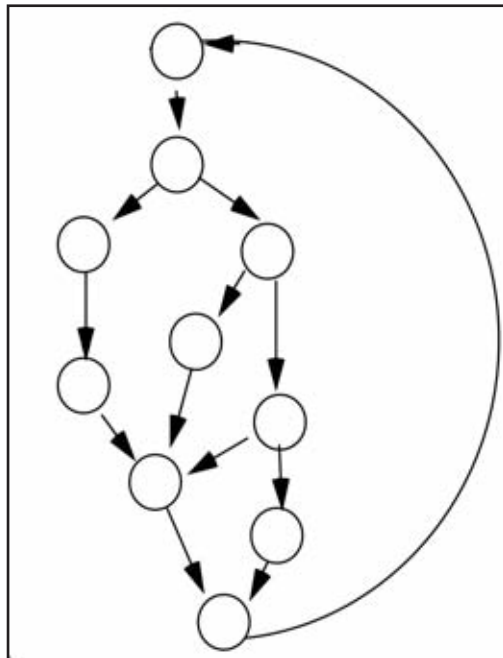


Fig. 19 Grafo fuertemente conectado, creado a partir del de la figura 18

Y sin embargo, como es fácil de comprobar, ambas tienen el mismo grafo, y por tanto la misma complejidad ciclomática. La idea de Myers [Myers, 1977] consiste en tener en cuenta estas diferencias, midiendo la complejidad como un intervalo, y no como un número. Para ello, toma como límite inferior del intervalo el número de sentencias de decisión más uno, y como superior, el número de condiciones individuales, también más uno.

El hecho de añadir un uno a la suma de sentencias o de condiciones está relacionado con la conveniencia de considerar grafos fuertemente conectados, como vimos al hablar del número ciclomático.

8. Medidas híbridas

Cada una de las métricas estudiadas hasta ahora tienden, por su propia construcción, a medir sólo cierta parte de los aspectos que contribuyen a que un programa sea complejo. Pero muchas veces nos interesa caracterizar de una forma más global la "complejidad de un programa". Para ello, han de considerarse a la vez varias propiedades del código. Una forma sencilla de hacer esto consiste en combinar de alguna forma métricas de diverso tipo, de las vistas hasta aquí. Es el caso de la métrica de Hansen (donde se utiliza una medida de flujo de control y otra de tamaño). También se puede diseñar una medida totalmente nueva, como ocurre con la métrica de Oviedo. Pasemos a estudiar estas dos.

8.1 Métrica de Hansen

Hansen [Hansen, 1978] propone una combinación del número ciclomático de McCabe con una medida del número de operandos. Concretamente, la métrica de Hansen es un par ordenado $(m1, m2)$, donde $m1$ y $m2$ se calculan como:

- $m1$ es el número de sentencias alternativas (IF, CASE, etc.) o iterativas (DO..WHILE, etc.).
- $m2$ es el número de operadores del programa, definidos de una forma similar al $n1$ de Halstead.

De esta manera tan sencilla, conseguimos caracterizar los fragmentos de código con dos números, que nos dan una idea de la complejidad de su flujo de control ($m1$) y de la cantidad de información total que contiene ($m2$). Además, al estar basada en dos medidas ampliamente estudiadas y probadas, puede asegurarse que la validez de la métrica de Hansen ha de ser bastante buena.

8.2 Métrica de Oviedo

En su propuesta, Oviedo [Oviedo, 1980] intenta medir simultáneamente la complejidad debida al flujo de datos y al flujo de control. Concretamente, propone la fórmula

$$C = a c_f + b d_f$$

donde c_f representa a la complejidad del flujo de control, d_f la del flujo de datos, y a , b son factores de peso, para dar más importancia a uno u otro de los dos aspectos medidos. En una primera aproximación pueden considerarse ambos iguales a 1.

El cálculo de c_f es sencillo a partir del grafo del programa. Es sencillamente igual al número de arcos que éste contiene.

Para estimar d_f hay que seguir un proceso algo más complicado. Tiene como base el concepto de "variable localmente expuesta". Este término se define, para un segmento de código, en nuestro caso cada uno de los nodos del grafo correspondiente. Son las variables cuyo valor es utilizado en ese segmento (en una asignación a otra variable, en una sentencia de salida, etc.), pero que lo han adquirido en otro anterior (en una asignación, sentencia de entrada, etc.). Pues bien, d_f se calcula a partir del número de posibles "adquisiciones" de valor que han podido tener las variables localmente expuestas de cada uno de los módulos.

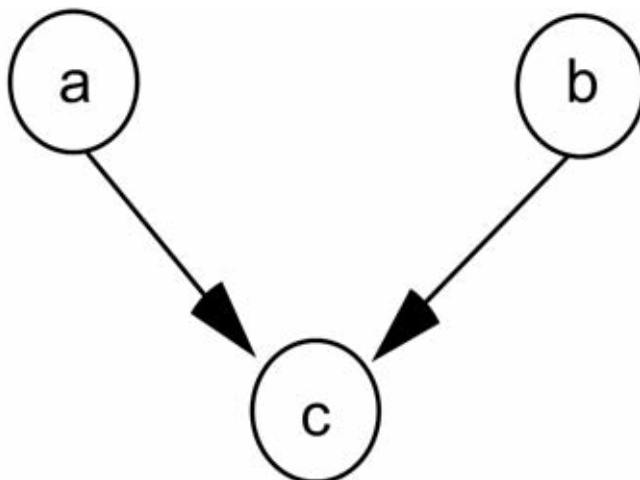


Fig. 20 Ejemplo de grafo que nos sirve para aclarar el concepto de exposición local de una variable.

Veamos un ejemplo que nos aclare el significado de la exposición local. Sea un programa que tenga un grafo como el de la figura 20, con tres nodos *a*, *b* y *c*. Y supongamos que se realizan las siguientes operaciones:

Nodo *a*:

PrePeras := 150;

PreUvas := 225;

KilPeras := 2;

KilUvas := 5;

Nodo *b*:

KilPeras := 3;

KilUvas := 4;

Nodo *c*:

KilPeras := 1;

PrecioCompra := KilUvas.PreUvas+KilPeras.PrePeras;

En el nodo *c*, serán variables expuestas localmente KilUvas, PreUvas y PrePeras (KilPeras no, ya que adquiere su valor en el mismo nodo). La c_f del nodo será $1+3=4$ (1 debido a la definición de KilUvas en el nodo *b*, y 3 debido a las asignaciones de PrePeras, PreUvas y KilUvas en el nodo *a*). Para los nodos *a* y *b*, c_f valdrá 0, ya que, según el grafo, no pueden tener variables localmente expuestas, pues no reciben arcos de ningún otro nodo. El c_f del total se calcula como la suma de los correspondientes a los nodos, y será por tanto $0+0+4=4$.

Si calculamos ahora la complejidad total para ese programa (teniendo en cuenta que en el grafo se observan 2 arcos, y por tanto $c_f=2$), tenemos $C=2+4=6$.

La métrica de Oviedo nos ofrece una valoración conjunta (con un solo número) de la complejidad debida al flujo de control y al flujo de datos.

9. Resumen

El software es algo indudablemente **complejo**. De este punto se deriva la gran cantidad de problemas que tenemos a la hora de construir programas sin errores, que cumplan en todo momento su funcionamiento previsto.

Es importante disponer de **herramientas** que nos permitan tratar con esta complejidad. En particular, es deseable disponer de procedimientos para medir algunos de sus aspectos, de cara a identificar programas especialmente complejos, para tratar de hacerlos más sencillos. También es importante **correlacionar** estas medidas con parámetros fundamentales del desarrollo software, como tiempo de desarrollo, de pruebas, dificultad de mantenimiento, número previsible de errores, etc.

Estas herramientas son las **métricas**. Según el aspecto en que se centren podemos clasificarlas en métricas de **tamaño**, de **estructura** y **flujo de datos**, de **flujo de control** e **híbridas**.

10. Bibliografía

Dividida en dos apartados: Notas Bibliográficas, donde se comentan algunos de los trabajos consultados en la elaboración de este capítulo, y Referencias Bibliográficas, donde pueden encontrarse todas las citas.

Notas bibliográficas

En el **artículo de Harrison, Magel, Kluczny y DeKock** [Harrison, Magel, Kluczny, y DeKock, 1982] se realiza una revisión de todas las métricas estudiadas aquí, y de algunas más, aunque sin gran detalle.

Para ampliar datos sobre cualquiera de las métricas de complejidad software que hemos tratado, puede acudirse a la referencia que se da en el apartado donde las estudiamos, que normalmente se refiere al trabajo original de su autor.

El número de **marzo de 1990** de la revista IEEE Software está dedicado a las métricas de software. En él pueden encontrarse artículos sobre recientes investigaciones sobre la aplicación de métricas a diversas etapas del ciclo de desarrollo de programas.

Referencias bibliográficas

Baird, J. C. y Noma, E. (1978): "**Fundamentals of scaling and psychophysics**", John Wiley and Sons, New York, pp.1-6., citado en [Harrison, Magel, Kluczny, y DeKock, 1982].

Basili, V. y Turner, A. (1975): "Iterative enhancement: a practical technique for software development", **IEEE Transactions on software engineering**, vol. SE-1, December, pp. 390-396, citado en [Harrison, Magel, Kluczny y DeKock, 1982].

Chapin, N. (1979): "A measure of software complexity", **Proceedings NCC**, pp.3-18], citado en [Harrison, Magel, Kluczny y DeKock, 1982].

Grady, R.B. (1990): "Work-product analysis: the philosopher's stone of software?", **IEEE Software**, March , pp.26-34.

Halstead, M. (1977): **Elements of software science**, Elsevier North-Holland, Nueva York, citado en [Harrison, Magel, Kluczny y DeKock, 1982]

Hansen, W. (1978): "Measurement of program complexity by the pair (Cyclomatic number, Operator count)", **ACM SIGPLAN Notices**, April, pp.29-33.

Harrison, W., Magel, K., Kluczny, R. y DeKock, A. (1982): "Applying software complexity metrics to program maintenance", **IEEE Computer**, September, pp.65-79.

Henry, S. y Selig, C. (1990): "Predicting source-code complexity at the design stage", **IEEE Software**, March, pp.36-44.

McCabe (1976): "A complexity measure", **IEEE Transactions on Software Engineering**, Vol. SE-2, December, pp.43-55.

Mills, H.D. y Dyson, P.B. (1990): "Using metrics to quantify development", **IEEE Software**, March, pp.15-16.

Myers, G. (1977): "An extension to the cyclomatic measure of program complexity", **ACM SIGPLAN Notices**, October, pp.61-64.

Oviedo, E. (1980): "Control flow, data flow and program complexity", **Proceedings COMPSAC 80**, pp.146-152.

Capítulo 15: El desarrollo del software

No disponemos de herramientas, ni siquiera de metodologías, que nos permitan transformar el software ordinario en otro que sea fiable y fácilmente mantenible. Los sistemas software medianamente grandes suelen estar "plagados" de errores, y realizar cambios en ellos es, cuando menos, una tarea arriesgada.

Frente a este duro panorama, nos encontramos con la necesidad de acometer el desarrollo de programas cada vez mayores. Para poder realizar estos desarrollos con la mejor calidad posible se hace necesaria la utilización de ciertas estrategias que, si bien no garantizan un buen resultado, si suelen mejorar bastante las características del producto desarrollado.



1. Introducción

Como puede leerse en [Grady, 1990], hoy por hoy no disponemos de herramientas, ni siquiera de metodologías, que nos permitan transformar el software ordinario en otro que sea fiable y fácilmente mantenible. En el campo del hardware, por el contrario, esta anhelada situación está mucho más cerca de la realidad. Así, disponemos de chips que son a la vez extremadamente complejos y muy fiables. Sin embargo, los sistemas software medianamente grandes suelen estar "plagados" de errores, y realizar cambios en ellos es, cuando menos, una tarea arriesgada.

Esta diferencia puede ser debida al hecho de que el desarrollo de hardware siempre ha estado constreñido por limitaciones físicas (por ejemplo, densidad de integración). Así, la evolución se ha hecho "paso a paso", añadiendo complejidad poco a poco en cada uno de estos pasos, a medida que se lograba introducir más componentes en una superficie dada. Pero el software no tiene este tipo de limitaciones, con lo que desde el principio tenemos una gran cantidad de complejidad, que hemos de manejar de alguna forma.

Por eso, el gran desafío con que se encuentra la gestión de proyectos software consiste precisamente en limitar los productos que se desarrollan en esos proyectos a unos niveles de complejidad aceptables y manejables. Dicho de otra forma, se pretende reducir los grados de libertad en la producción de software para, al operar dentro de unos ciertos márgenes, mantener la complejidad resultante lo más baja posible.

Esto ha llevado a la concepción y uso de varios modelos del ciclo de vida. Con ellos se intenta descomponer los problemas de la gestión del proyecto de forma lógica, a la vez que generar productos tras cada etapa del modelo. Estos productos pueden ser usados para comprobar si estamos moviéndonos en la dirección deseada, o si por el contrario nos apartamos de los objetivos de complejidad previstos. Al fin y al cabo, utilizamos la acreditada técnica del "divide y vencerás".

Para enmarcar el estudio de los problemas relacionados con el desarrollo de software, señalemos que estamos tratando con uno de los llamados sistemas antropotécnicos, dentro del modelo de tres niveles de complejidad de Sáez Vacas (véase el capítulo sobre **Marcos Conceptuales**). El lector estará de acuerdo con esta afirmación si piensa que el proceso de desarrollo de programas un poco grandes implica la gestión y coordinación de los esfuerzos de numerosos grupos de personas, ayudadas de herramientas tecnológicas cada vez más avanzadas.

2. El ciclo de vida

En principio, el ciclo de vida de un proyecto software incluye todas las acciones que se realizan sobre él desde que se especifican las características que debe tener, hasta que se mantiene en operación. A veces (aunque no será éste nuestro caso) se incluyen en el ciclo de vida las modificaciones que pueden realizarse al sistema para adaptarse a nuevas especificaciones.

Podría pensarse que el ciclo de vida de un programa no tiene por qué seguir un desarrollo "lineal", entendiéndolo como tal una sucesión de etapas. En principio, las distintas actividades que se realizan son bastante independientes, y pueden llevarse (hasta cierto punto) en paralelo. Por ejemplo, para empezar a codificar hay que tener mínimamente claras las especificaciones que hay que cumplir. Pero (aunque no es una buena decisión, como veremos más adelante), podría pensarse en comenzar la producción de código mientras se completan las especificaciones, para poder irlo probando, por ejemplo. Más adelante se harían las modificaciones necesarias.

Pero si el desarrollo de productos software ya es algo complejo en sí mismo (véase el capítulo sobre Medidas o **Métricas de la Complejidad del Software**), aún lo complicaremos más si intentamos "hacerlo todo a la vez", sin seguir una cuidadosa y detallada planificación. Y esto es precisamente lo que pretenden los modelos del ciclo de vida del software: simplificar en lo posible la gestión del proceso de desarrollo. La meta está en añadir la mínima complejidad que sea posible a la que de por sí ya implica el software.

Desde el punto de vista del esquema HxIxO->IO, podríamos decir que los modelos del ciclo de vida son un instrumento conceptual (I) que permite a la persona encargada (H) de gestionar un desarrollo de software (el O será por tanto el propio proceso de desarrollo) tratar con un problema más sencillo (el IO resultante).

Para ello, estos modelos dividen el proceso de desarrollo en unas cuantas etapas bien diferenciadas, y definen los posibles caminos por los que se deben relacionar. Además intentan que estos caminos lleven a un "progreso lineal", en el sentido de que antes de comenzar una etapa se haya concluido exitosamente (con las especificaciones cumplidas) la anterior. Sin embargo, esto no es siempre posible, y hay que recurrir a iteraciones (por ejemplo, entre el diseño y la codificación), que nos lleven mediante aproximaciones sucesivas a cumplir con los objetivos de la mejor forma posible.

Desde el punto de vista jerárquico (véase el capítulo sobre las **Jerarquías**) esta división en etapas puede verse como una jerarquía multicapa de toma de decisiones. Así, cada una de las etapas (capa de decisiones) termina cuando, tras haber hecho todas las elecciones necesarias, se han cumplido los objetivos marcados, sentando las bases para la siguiente etapa. Al dividirse el problema en estas capas, en cada momento del desarrollo nos enfrentamos con una complejidad menor (únicamente la debida a cada capa, ya que las anteriores habrán sido satisfactoriamente resueltas).

3. El modelo de desarrollo en cascada

Uno de estos modelos del ciclo de vida, quizás el más ampliamente utilizado, es el del desarrollo en cascada. En él, cada etapa deja el camino preparado para la siguiente, de forma que esta última no debe comenzar hasta que no ha acabado aquella. De esta forma, se reduce mucho la complejidad de la gestión, ya que basta con no dar por terminada una etapa hasta que haya cumplido totalmente con sus objetivos. De esta forma, la siguiente puede apoyarse con total confianza en ella. A la hora, por ejemplo, de fijar plazos, se podrían establecer planes de una forma totalmente secuencial, quedando perfectamente delimitadas las responsabilidades de los equipos que desarrollen cada etapa.

En la realidad la aplicación de este modelo no suele ser tan radical. Aunque se intenta conseguir la mayor secuencialidad posible, es difícil evitar las "vueltas atrás". Si después de la terminación de alguna etapa los resultados no son los esperados, en la práctica es muy posible que el problema esté en la mala realización de una etapa anterior. Y esto es así porque no sabemos cómo decidir con total certidumbre que una etapa ha sido perfectamente desarrollada hasta que se observan las consecuencias, quizás varias etapas y bastante tiempo después de que fue "cerrada". En estos casos, habrá que volver a ella, refinando el producto de una forma iterativa hasta que se considere que tiene la calidad deseada.

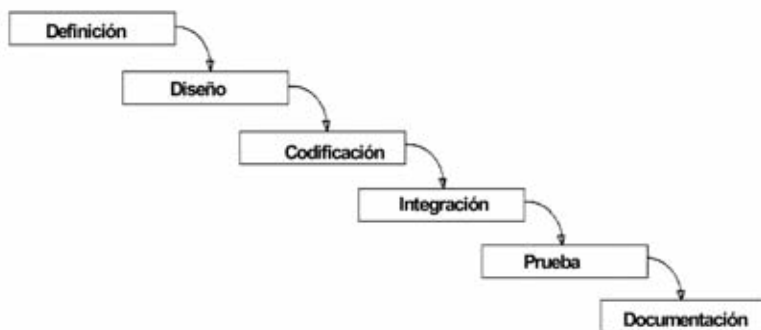


Fig. 1 Modelo en cascada del desarrollo de software

En el modelo "puro", las fases en que se suele dividir el ciclo de vida en este modelo son [Grady, 1990]:

- a. Definición (análisis de los requerimientos software).
- b. Diseño (podría dividirse en preliminar y detallado).
- c. Codificación.
- d. Integración.
- e. Prueba.
- f. Documentación.

Estas fases de desarrollarían una tras otra, excepto quizás las dos últimas. La prueba de módulos podría realizarse después de la codificación y la del sistema completo tras la integración. La documentación, por su parte, puede irse creando a lo largo de todo el proceso.

Sin embargo, los caminos reales que se siguen en el desarrollo de software suelen parecerse mucho más a los que se pueden ver en la figura 2 (basada en [Fox, 1982]). En ella, las flechas que apuntan en sentido descendente representarían el modelo puro, mientras que las ascendentes corresponden a los demás caminos que se suelen seguir en la realidad.

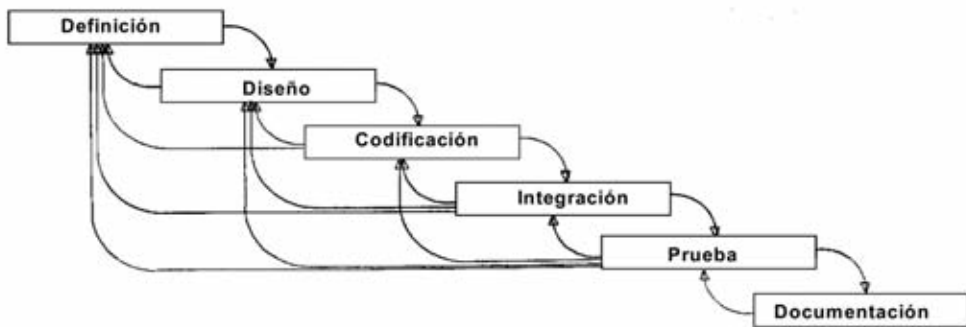


Fig. 2 Caminos reales en el desarrollo de software [Fox, 1982]

Pasemos a describir ahora cada una de las etapas del modelo en cascada, que ya hemos nombrado.

3.1. Definición

La definición de requisitos o especificación de características que ha de cumplir el software que vamos a desarrollar es la primera etapa del modelo en cascada. Y probablemente sea la más importante. Al fin y al cabo, lo que sea o no sea el producto final depende de decisiones tomadas en esta etapa. Se trata fundamentalmente de estudiar las necesidades y preferencias del usuario. Es también muy importante dejar clara constancia de las decisiones tomadas en esta etapa, para ser tenidas en cuenta posteriormente. Por ello, la documentación producida en esta fase debe ser concreta y estar siempre disponible durante el resto del proceso.

Pero, normalmente, nuestro software no será mas que una parte de un sistema mayor. Y puede ser que "herede" problemas de indefinición de este sistema. Por ejemplo, como el hardware es muy difícil de modificar, a menudo los programas sufren cambios de última hora para "tapar" sus defectos. Esto hace que, en un proyecto real, sean bastante normales los caminos de vuelta desde etapas posteriores a ésta de definición (ver figura 2).

Por si hubiera pocos problemas debemos tener en cuenta que el entorno en el que nos movemos suele ser muy variable, de forma que las características que piden los usuarios suelen cambiar muy rápidamente con el tiempo. Además, las innovaciones tecnológicas hacen posible modificar ciertas partes del sistema para ganar, por ejemplo en eficiencia. Por todo esto, la facilidad de modificación del software resultante es siempre un requisito fundamental, que debe compaginarse con los demás.

La descomposición en niveles de abstracción es una metodología que nos puede ayudar a abordar los problemas que aparecen en esta etapa. Las características que debe tener esta descomposición son las habituales en el análisis de sistemas complejos (ver capítulo sobre las **Jerarquías**): pocos elementos en cada nivel de abstracción, contextos limitados y bien definidos, etc.

De un tiempo a esta parte se están comenzando a utilizar técnicas formales de definición. Esto permite generar especificaciones coherentes y sin ambigüedades. Además se está investigando activamente en la generación automática de software a partir de definiciones escritas en lenguajes formales (aunque aún no se han logrado grandes resultados prácticos en este campo).

3.2. Diseño

Una vez planteada la especificación del programa, hay que analizar desde un punto de vista técnico las posibles soluciones. Entre ellas, se elegirá la que se considere más adecuada. A partir de ese momento, se decidirá la estructura general del programa (subdivisión en partes y relaciones entre ellas). Para cada una de las partes se seguirá recursivamente un proceso similar, hasta que tengamos totalmente definido el programa y estemos listos para pasar a la fase de codificación.

En el análisis de cada una de las partes nos encontraremos normalmente con que hay varias soluciones posibles (por ejemplo varios algoritmos para realizar la misma tarea). La elección de una de ellas suele realizarse de una forma más o menos intuitiva: no hay metodologías efectivas que nos ayuden en esta decisión.

Como puede deducirse de lo dicho hasta aquí, la descomposición en niveles de abstracción también será útil en esta fase. Cada etapa del proceso recursivo descrito puede constituir un nivel de abstracción. Si, además, utilizamos las posibilidades de ocultación de información que nos permite esta metodología, podremos descomponer nuestro programa en pequeños módulos fáciles de modificar.

En el nivel más bajo del diseño hay que decidir la estructura de control y el flujo de datos del módulo. El uso de la programación estructurada facilita enormemente la comprensión de los algoritmos, al limitar los flujos de control posibles.

El producto final de la etapa de diseño puede ser un organigrama, unas líneas de pseudocódigo, etc. Algunos lenguajes de programación (como Ada) permiten hasta cierto punto realizar el diseño en el propio lenguaje, y compilarlo posteriormente. Así pueden detectarse incoherencias y ambigüedades de una forma automática. Además se favorece en gran medida la integración con la etapa de codificación.

Algunos trabajos recientes ([Rombach, 1990], [Henry y Selig, 1990]) proponen utilizar métricas en la fase de diseño para predecir la calidad del producto software antes de llegar a la codificación. Así se ahorrarían esfuerzos, al encontrar pronto zonas de gran complejidad y de poca calidad. De esta forma estas zonas podrían rediseñarse, consiguiéndose así que den menos problemas en posteriores etapas del desarrollo.

3.3. Codificación

En un proyecto grande ésta es la etapa más sencilla (en contra de lo que suele suponer cualquier persona que comienza a aprender un lenguaje de programación). Si el diseño es adecuado y suficientemente detallado la codificación de cada módulo es algo casi automático.

Una de las principales decisiones a tomar en esta fase es la del lenguaje a emplear, aunque a veces en el diseño ya está de alguna forma implícito. Desde hace tiempo la tendencia es a utilizar lenguajes de más alto nivel, sobre todo a medida de que se dispone de compiladores más eficientes. Esto ayuda a los programadores a pensar más cerca de su propio nivel que del de la máquina, y la productividad suele mejorarse. Como contrapartida este tipo de lenguajes son más difíciles de aprender. Y además hay que tener en cuenta que los programadores suelen ser conservadores y reacios a aprender nuevos lenguajes: prefieren usar los que ya conocen. La existencia, en una organización, de una gran cantidad de programas desarrollados en un determinado lenguaje, hace además muy dura la decisión de cambiar a uno nuevo.

Evaluar la calidad de la codificación es una tarea nada fácil. Para un mismo diseño son posibles muchas implementaciones diferentes. Y no hay criterios claros que nos permitan decidir cuál es la mejor. En este punto, las métricas del software pueden ser utilizadas en nuestra ayuda (ver capítulo sobre las **Métricas**).

Cuando intervienen varias personas, pueden aparecer problemas a la hora de realizar modificaciones, debido a que cada uno tiene su propio estilo. Por eso se hace necesario definir estándares de estilo para facilitar la legibilidad y claridad del software producido.

3.4. Integración

Una vez que tenemos los módulos codificados, hay que ensamblarlos. Desgraciadamente el proceso no consiste simplemente en unir piezas. Suelen aparecer problemas con las interfaces entre los módulos, con la comunicación de datos compartidos, con el encadenamiento de flujos de ejecución, etc.

Si el programa es además bastante grande, la gestión de versiones se convierte en un problema no despreciable. Afortunadamente, ésta es una de las etapas donde disponemos de más herramientas CASE, que nos pueden ayudar.

3.5. Prueba

En esta fase hay que comprobar que las especificaciones se cumplen perfectamente y en todos los casos. En la realidad es prácticamente imposible probar un programa totalmente: por ello siempre suele quedar algún error escondido. Este problema se agrava cuando sobre él se realizan repetidos cambios y correcciones. Si no los gestionamos de un forma adecuada podemos acabar con un conjunto de parches que más que soluciones aportan problemas.

Actualmente se están comenzando a utilizar técnicas de verificación y validación como alternativa a la simple prueba de programas. Según Wallace y Fujii [Wallace y Fujii, 1989], la verificación y validación es una disciplina de ingeniería de sistemas, que intenta evaluar el software desde un punto de vista sistémico. Utiliza una aproximación estructurada para analizar y probar el software en relación con todos los aspectos del sistema en el cual se incluye, y en especial con el hardware, los usuarios y las interfaces con otras piezas de software.

Idealmente, la verificación y validación se realiza paralelamente al desarrollo de software, durante todo su ciclo de vida (por lo que no entra en el modelo en cascada, estrictamente hablando), y pretende alcanzar los siguientes objetivos:

- a. Descubrir pronto errores de alto riesgo, dando al equipo de diseño la oportunidad de elaborar una solución adecuada, evitando que se vea obligado a poner un "parche" si el error se detecta demasiado tarde.
- b. Evaluar el ajuste de los productos desarrollados a las especificaciones del sistema.
- c. Proporcionar al equipo de gestión información actualizada sobre la calidad y el progreso del esfuerzo de desarrollo.

Éste de la verificación y validación es un campo donde se están realizando activas investigaciones, mientras comienzan a obtenerse los primeros frutos.

3.6. Documentación

La documentación es algo totalmente necesario para poder mantener un programa. Incluso la persona que lo ha codificado se perderá con gran facilidad en un programa a los pocos meses de haberlo terminado. No sólo hay que documentar el código (las conocidas líneas de comentario del programa), sino todas las etapas del ciclo de vida. Especialmente es importante que todas las decisiones que se han tomado queden claramente expuestas, así como las razones que han llevado a ellas.

Además, hay que generar la documentación de "caja negra", esto es, la que se refiere no a aspectos internos del programa, sino a su manejo y características "externas". Esto incluye normalmente un manual de usuario, para las personas que normalmente van a utilizarlo (en el caso de que sea un programa directamente utilizado por personas) y un manual de referencia técnica, donde se dan detalles de su instalación y explotación, de cara al personal técnico encargado de estas tareas.

En el modelo en cascada hemos colocado la etapa de documentación al final, porque es cuando se realizará la documentación definitiva, y especialmente los manuales "de caja negra" de los que hemos hablado. Pero es conveniente ir preparándola a lo largo de todo el desarrollo, según van realizándose las actividades a documentar.

Para gestionar esta etapa (llevar el control de las versiones de la documentación, incluso generarla automáticamente en algunos casos) también se dispone de herramientas informáticas de ayuda.

4. Los "productos intermedios"

Tras cada una de las etapas del ciclo de vida se genera, como resultado final, algún tipo de producto. Son lo que llamaremos "productos intermedios". Estos productos constituyen la base del trabajo de la siguiente etapa. Por ejemplo, a partir del pseudocódigo obtenido en la fase de diseño, los codificadores escribirán el programa. Y este programa (resultado de la etapa de codificación) será la base para la integración. Una lista más exhaustiva de los productos intermedios que se obtienen en cada etapa del modelo en cascada puede verse en la figura 3.

Pero estos productos pueden usarse para algo más que meramente como apoyo de la fase siguiente. Según Grady [Grady, 1990], una correcta utilización de los productos intermedios ayuda a producir software de calidad, ya que:

- a. Cada producto intermedio suele seguir alguna forma de representación estándar que garantiza un cierto grado de terminología común.

- b. Existen herramientas que pueden aplicarse a estos productos, para hacer comprobaciones sobre ellos, aportando así realimentación inmediata a los ingenieros de desarrollo (generalmente mediante la forma de avisos y mensajes de error).
- c. La terminología común simplifica las inspecciones por parte de otros equipos de trabajo. Así se facilita la detección de errores que las herramientas automáticas no son capaces de detectar.
- d. También pueden utilizarse herramientas que calculen ciertas métricas sobre diversos aspectos de la complejidad de los productos intermedios. Así se pueden detectar zonas con mayor probabilidad de que presenten errores, o que tengan un difícil mantenimiento.



Fig. 3 Productos que se generan en cada una de las etapas de producción de software (según el modelo de la cascada), basado en [Grady, 1990]

La idea central de este aprovechamiento de los productos intermedios del desarrollo software reside en la información que éstos aportan, como resumen final que son de su etapa correspondiente. Así, su análisis permite una realimentación rápida y concreta, y una forma de ir midiendo paso a paso la calidad del software que se está produciendo.

5. Resumen

En este capítulo hemos tratado de exponer los problemas que aparecen en el desarrollo de sistemas software reales, y de los enfoques que se utilizan para abordarlos.

Comenzamos introduciendo el tipo de sistemas con el que tratamos (sistemas antropotécnicos), y las diferentes características que los hacen complejos y difíciles de abordar. Luego hemos hablado de una de las soluciones más utilizadas: la reducción de complejidad mediante la limitación de los grados de libertad del sistema. Y de cómo se concreta esta solución: el modelo en cascada del ciclo de vida de un programa.

Y por último, nos hemos detenido en las etapas que constituyen este modelo en cascada, poniendo de relieve los aspectos que nos han parecido más destacables en cada una de ellas. Es uno de los modelos posibles. Hay otros muchos, como el modelo en espiral, pero en cualquiera de ellos se hace patente que la complejidad surge de la necesidad de coordinar económicamente un elevado número de personas, técnicas y equipo. Es la complejidad de gestión del software.

6. Bibliografía

Dividida en dos partes. En primer lugar, Notas Bibliográficas, donde se describen los trabajos consultados más relevantes sobre el tema. Después, Referencias Bibliográficas, donde pueden encontrarse todas las citas utilizadas en el capítulo.

Notas bibliográficas

Un trabajo muy utilizado ha sido el de Grady [Grady, 1990], que entre otras aportaciones originales, incluye el análisis de los productos intermedios del ciclo de vida como ayuda para la producción de programas de calidad.

También es necesario citar el número de mayo de 1989 de la revista IEEE Software, dedicado a la verificación y validación, de donde están tomadas las ideas sobre este particular que pueden encontrarse en el capítulo.

El libro de Fox [Fox, 1982] puede ser de gran utilidad para ampliar conocimientos sobre las etapas del ciclo de vida de un sistema software, y su significado.

Referencias bibliográficas

Fox, J.M. (1982): "**Software and its development**", Ed. Prentice-Hall.

Grady, R.B. (1990): "Work-product analysis: the philosopher's stone of software?", **IEEE Software**, March, pag.26-34.

Henry,S. y Selig, C. (1990): "Predicting source code complexity at the design stage", **IEEE Software**, March, pag.36-45.

Rombach, H.D. (1990): "Design measurement: some lessons learned", **IEEE Software**, March , pag.17-25.

Wallace,D.R. y Fujii,R.U. (1989): "Software verification and validation: an overview", **IEEE Software**, May. pag.10-17.

Bloque IV: Anexos

Capítulo 16: Iniciativa de Defensa Estratégica (S.D.I.)

Este texto ha sido redactado por Gustavo Alonso en 1990 para el curso de Cibernética y Teoría de Sistemas de la Escuela Técnica Superior de Ingenieros de Telecomunicación de Madrid. Se basa en su mayor parte en artículos del IEEE; para evitar las referencias continuas a lo largo del texto, incluimos aquí los números utilizados para desarrollar este tema:

IEEE Spectrum, Septiembre 1985, número monográfico sobre la S.D.I.

IEEE Computer, Noviembre 1986, "Can S.D.I. Software be error free?", Ware Myers, pp. 61-67.

IEEE Spectrum, Noviembre 1988, número monográfico dedicado al coste de los proyectos de defensa.

IEEE Spectrum, Marzo 1989, número monográfico dedicado a la repercusión en la tecnología de los tratados de paz y de desarme.

IEEE Spectrum, Noviembre 1989, "S.D.I. in transition", John A. Adam, pp. 32-38.

1. Introducción

El presente capítulo pretende ser un ejemplo donde aplicar algunas de las ideas que se han visto a lo largo de todos estas Notas. No se pretende hacer ningún juicio de valor sobre el tema si no es desde un punto de vista estrictamente tecnológico y desde la complejidad. Evidentemente la Iniciativa de Defensa Estratégica (S.D.I.) es un tema controvertido, quizá ahora no tanto como lo fue en su momento pero, en cualquier caso, ejemplifica magníficamente varias de las ideas que hemos tratado.

El proyecto de la S.D.I. es de una complejidad formidable y los objetivos que persigue en un amplio abanico de tecnologías son muy ambiciosos. Aún está por ver que pueda manejarse toda esa complejidad. Como veremos, al tratar los problemas que plantea este proyecto nos iremos encontrando con algunos de los problemas que ya hemos estudiado referentes al software y al hardware, además de muchos otros, y con muchas de las propuestas conceptuales que se hacían en la primera parte (niveles, jerarquías, sistemas, incertidumbre, etc.). Lo que tiene de atractivo la S.D.I., desde el punto de vista de nuestro estudio, es que intenta abordar todos esos problemas en un único proyecto, siendo un buen campo teórico donde ejercitar las ideas de complejidad.

Una forma de ver todo el proyecto es a través del modelo de tres niveles de complejidad de Sáez Vacas (referencia a **Marcos Conceptuales**) y es la que hemos escogido para desarrollar brevemente aquí. Pero no es, en absoluto, la única. Instamos al lector a que intente ver todas las propuestas de la S.D.I. a la luz de diversos modelos y definiciones de complejidad y comparar los resultados que vaya obteniendo con lo visto hasta ahora referente a tecnología. Algunas ideas pueden ser la complejidad relacionada con la incertidumbre (principios de Klir), la simplificación, jerarquías de niveles, definición amplia de complejidad, etc.

2. ¿Qué es la Iniciativa de Defensa Estratégica?

En este primer apartado vamos a tratar de explicar los supuestos de partida de la S.D.I., qué pretende conseguir y cómo. De todo ello, lo que más nos interesa es cómo este proyecto afecta a la innovación tecnológica al plantear una serie de cuestiones críticas, desde el punto de vista puramente tecnológico, que sin duda afectan al desarrollo y a la innovación. Más adelante profundizaremos en la problemática relacionada con el software, aquí pretendemos destacar la gran demanda de tecnología en todos los campos que supone un proyecto como éste.

2.1 Planteamiento básico

El proyecto de la S.D.I. (Strategic Defense Initiative) tiene su origen más inmediato en un discurso de Ronald Reagan, pronunciado el 23 de Marzo de 1983, en el que proponía convertir las armas nucleares en algo "impotente y obsoleto" a través de la construcción de un sistema de defensa capaz de inutilizar misiles nucleares cuando se lanzan contra un objetivo.

Desde entonces, la S.D.I. ha levantado apasionadas controversias tanto en el campo político como en el tecnológico y las interpretaciones de los objetivos que se persiguen son de lo más variado. Quizá la más conocida es la de la "Guerra de las Galaxias" que tiene su origen en uno de los objetivos concretos que persigue la S.D.I., como es la instalación en el espacio de bases antimisil destinadas a derribar en vuelo las armas nucleares lanzadas contra los EEUU. También se conoce a la S.D.I. como "escudo", centrándose en la idea de que de lo que se trata es de crear una barrera capaz de detectar y detener ataques nucleares. Ambos términos son demasiado simplistas y llevan a confusión, uno por considerar un único aspecto del problema, aunque sea ciertamente espectacular, y otro porque parte de un presupuesto erróneo, ya que la S.D.I. no es perfecta y la cobertura no será completa.

La interpretación que hace el Pentágono también es interesante, en lugar de la destrucción mutua asegurada (Mutually Assured Destruction, MAD) se pretende llegar a una situación de defensa mutua asegurada.

En pocas palabras, la S.D.I. pretende desarrollar un sistema de defensa, que cubra todo el territorio de los EEUU, encargado de detectar y eliminar las armas nucleares lanzadas contra este territorio.

2.2 Diseño inicial de la S.D.I.

La S.D.I parte de una aproximación por capas al problema de la defensa. Cada una de estas capas está relacionada con alguna de las diferentes fases del lanzamiento de un misil (lanzamiento, postlanzamiento, curso intermedio y fase terminal). El éxito depende, en gran medida, de la capacidad para destruir el mayor número de misiles que sea posible en la fase de lanzamiento (desde que despegan hasta que se dejan libres en la atmósfera). Los misiles que atraviesen este primer sistema de defensa han de ser destruidos en las fases posteriores intentando siempre reducir al mínimo el número de cabezas nucleares que entran en la atmósfera.

Cada una de estas fases y cada tipo de misil requiere un armamento diferente y, a medida que aumenta la complejidad del problema, más exótico. En la fase de lanzamiento hay que distinguir dos tipos de cohetes, los de lanzamiento rápido (que en lanzamientos intercontinentales tardan 1 minuto en superar esta fase) y los de lanzamiento lento (que en misiles intercontinentales supone unos 3 minutos en esta fase). Para derribar los cohetes de lanzamiento lento hay que disponer de bases espaciales de misiles nucleares, para los de lanzamiento rápido se intentan desarrollar láseres disparados también desde bases espaciales. Este mismo tipo de armas, siempre desde bases en el espacio, junto con haces de partículas son las que se utilizan para atacar la segunda fase, la de postlanzamiento.

En esta segunda fase, los cohetes se separan en varias cabezas nucleares y en diversos señuelos, lo cual añade más complejidad al problema pues hay que distinguir entre los objetivos reales (cabezas nucleares) y los señuelos. En la tercera fase, durante el vuelo fuera de la atmósfera que dura de 6 a 22 minutos, se utilizan pequeños proyectiles lanzados desde tierra junto con haces de partículas. En la fase terminal se emplean proyectiles de alta velocidad (armas de energía cinética) y cohetes para destruir las cabezas nucleares restantes.

Al contrario de lo que puede parecer, el desarrollo de este tipo de armas no nucleares es el campo más maduro de la S.D.I. y muchas de las tecnologías no son nuevas (excepto los láseres de rayos X). Además aparecen consideraciones nuevas como que es más barato desarrollar estas armas de defensa para destruir misiles que hacer nuevos misiles de ataque.

La utilización de este tipo de armas exige avances tecnológicos muy importantes, pues algunas de ellas han de estar en bases espaciales y las que se disparan desde tierra han de poder guiarse hasta su objetivo. Las propias armas son en sí tecnología punta: tanto las armas de energía directa (láseres y haces de partículas) como las armas de energía cinética (cohetes y rampas de lanzamiento).

Efectividad de la S.D.I.

Un posible planteamiento estratégico del sistema a construir puede ser el siguiente:

Ataque

1.400 Misiles Balísticos Intercontinentales, ICBM

x 10 cabezas nucleares/misil = 14.000 cabezas

x 10 señuelos/cabeza nuclear = 140.000 señuelos

Defensa

**** Fase de lanzamiento***

5 satélites de vigilancia, 160 estaciones láser de batalla y radar, con una efectividad del 90%

14.000 x 0,1 = 1.400 cabezas nucleares

140.000 x 0,1 = 14.000 señuelos

**** Fase de curso intermedio***

28.000 armas de energía cinética, 20 satélites de vigilancia, con una efectividad del 90%

1.400 x 0,1 = 140 cabezas nucleares

14.000 x 0,1 = 1.400 señuelos

**** Fase terminal***

75 radares, 25 aviones de vigilancia, 140.000 cohetes de energía cinética, con una efectividad del 90%

14 cabezas nucleares alcanzan su objetivo

2.3. Innovación Tecnológica en la S.D.I.

La Iniciativa de Defensa Estratégica es un enorme reto a la ingeniería y la ciencia. Los problemas que plantea un escenario como el descrito en el apartado anterior son muy complejos y comprenden un abanico muy amplio de tecnologías que han de trabajar de forma integrada. Muestra de ello son los ocho programas en los que se subdividió la S.D.I. a finales de 1983: sistemas; sensores; vigilancia, ataque y tecnologías clave; armas de energía cinética; armas de energía directa; innovación científica y tecnológica; y asuntos externos.

El proyecto de la S.D.I supone una fuerte inversión en investigación y desarrollo mantenida durante bastantes años. Dada la creciente dependencia que tiene la investigación estadounidense del presupuesto de defensa, no son pocos los que defienden que además del problema de la fiabilidad de la S.D.I., han de considerarse todos los avances que se pueden lograr con estas inversiones. Orquestando los mecanismos necesarios de transferencia tecnológica la sociedad puede beneficiarse de los resultados de estas investigaciones.

Esta parece la interpretación de algunos países europeos y gran parte de las compañías implicadas. Un ejemplo de este interés por los proyectos de investigación generados es el de Dinamarca, cuyo parlamento se opone rotundamente a la militarización del espacio pero al mismo tiempo ha dejado bien claro que esto es sólo una resolución, no una ley, y que por tanto las compañías danesas participarán en el proyecto. Está claro que existe un riesgo enorme de "perder el tren" de la tecnología si no se participa en la investigación y desarrollo de estos sistemas.

El problema con que se enfrenta está lejos de ser sencillo y se han señalado cuatro como los obstáculos principales con los que se va a encontrar:

1. Distinguir las cabezas nucleares de los señuelos (según algunos cálculos un ataque consistiría en unos 3.000 misiles con un total de unas 30.000 cabezas nucleares y 250.000 señuelos).
2. Diseñar software fiable para gestionar las operaciones (se tendrán que tomar cientos de miles de decisiones independientes y sincronizarlas en cuestión de milisegundos).
3. El sistema ha de ser suficientemente robusto como para poder soportar un ataque contra él y seguir funcionando.
4. Ha de ser un sistema suficientemente barato como para que el enemigo no se sienta tentado a superarlo aumentando su capacidad ofensiva.

El primer punto obligará a mejoras importantes en los sensores, radares y mecanismos de detección pues son una parte crucial del sistema. Muchas de las armas empleadas en la S.D.I. requieren una gran cantidad de energía y no se puede desperdiciar disparando contra objetos que no son una amenaza, por otro lado, los sensores que descartan objetivos identificándolos como señuelos se enfrentan a la posibilidad de que haya cabezas nucleares que intentan pasar por señuelos, con lo que la labor de identificación es aún más compleja. A esto se añade el problema de que hay que utilizar la energía exacta, no se puede disparar con

menos energía porque no se puede creer que se ha eliminado algo que sigue siendo una amenaza y tampoco se puede derrochar energía aumentando las dosis cuando esto no es necesario.

En 1984 se hizo un experimento en el que se informó que se había conseguido detectar, en el espacio, el calor de un cuerpo humano a 1.600 km. de distancia. Un proyecto como el de la S.D.I. exige poder detectar objetivos entre 20 y 50 grados Kelvin más fríos que la silueta de un cuerpo humano contra el calor de la superficie de la tierra a distancias superiores a los 5.000 km.

Otro punto muy interesante es el número de estaciones necesarias para frenar un ataque. Sobre esto no hay acuerdo pues depende de las suposiciones de partida que se hagan y consideraciones técnicas muy complejas (potencia de los láseres, difusión de éstos, número de misiles enemigos, altitud de las bases, etc.), pero las simulaciones realizadas para un ataque de 1.400 misiles van desde 30 estaciones a 2.400 y proponen desde un crecimiento lineal de las bases con el número de misiles ofensivos a un crecimiento del 40% cuando el ataque se dobla. Además, estas simulaciones han aportado otros datos interesantes como la velocidad de eliminación de misiles en la fase de lanzamiento (10 por segundo) o el tiempo que hay enfocar un misil con un láser para destruirlo (desde 1,7 a 6,6 segundos, dependiendo de la distancia).

Esto nos lleva a la situación actual de los láseres, los haces de partículas y la capacidad de potencia en las bases espaciales. Con datos de 1985 esta tabla resume muy bien las diferencias entre lo que se tiene y lo que se persigue:

	Objetivos	Actualmente
Brillo del láser	10 ²¹ watos/esteroradian	10 ¹⁷ W/sr
Energía de partícula	250 megaelectronvoltios	50 MeV
Potencia en el espacio	100-300 megawatios	25 kW
Eficiencia	600 Kilojulios por Kg.	-
Coste	1 dólar por Kilojulio	50 dólares/Kj

Las armas de las que antes hablábamos requieren unas condiciones muy especiales. Los láseres y haces de partículas necesitan mucha energía para ser efectivos y esta energía se la ha de proporcionar la base espacial en la que se instalen. Los láseres, por ejemplo, necesitan unos 100 MW, las rampas de lanzamiento de proyectiles de alta velocidad necesitan pulsos de energía del orden de los gigawatios. Esto significa que hay que mandar mucha masa al espacio y se traduce en un coste muy elevado. Por ello, otro de los objetivos de la S.D.I. es reducir el coste de acceder al espacio, en 1985 este coste era de 1.200 dólares cada libra de peso y se quiere reducir a 10 dólares por libra de peso.

	Objetivos	Actualmente
Potencia básica	0,1-0,2 MW durante 7 años	12 kW durante 4-5 años
Potencia en alertas	2.0-20 MW tiempo indefinido	12 kW durante 4-5 años
Potencia en combate	100-300 MW durante 30 minutos	Ninguna
Condiciones de potencia	100 kV, 1000-3000 A	300 V, 10 A
Eficiencia	3 kW/kg	0,005 kW/kg

Las armas de energía cinética plantean problemas muy interesantes de resistencia de materiales. Los proyectiles se han de disparar a una velocidad superior a los 20 km/s, a esa velocidad las "balas" llegan a fundir el cañón que las dispara, lo que obliga a buscar materiales especiales que soporten esa presión y temperatura. En un experimento en el que se consiguió una velocidad de 11 km/s con un proyectil de 3 gramos, la presión ejercida hizo que el cañón se deshiciera a medida que la bala pasaba por él, el proyectil salió gracias a que su velocidad era tan alta que se mantenía por delante de la onda de choque que destruyó el cañón. Además, el tipo de proyectiles que se quieren conseguir, de unos 3 kg., lanzados a una velocidad de entre 15 y 30 km/s, requieren un pulso de potencia de varios gigawatios, potencia que ha de ser proporcionada por una fuente suficientemente pequeña como para poder ser instalada en el espacio.

Todos estos ejemplos sólo son una muestra de la gran cantidad de avances tecnológicos de importancia que exige desarrollar un sistema como el que propone la S.D.I., y no sólo en armas o en sensores, sino también investigación en fuentes de potencia (química, nuclear, solar, por radioisótopos, etc.), circuitos eléctricos y electrónicos (capaces de soportar esa potencia en un medio extremadamente hostil y con unas limitaciones grandes), aviónica (que en el caso de los proyectiles de

energía cinética tendrán que soportar aceleraciones de 100.000 G), ordenadores ópticos (para conseguir la velocidad de proceso adecuada), comunicaciones en el espacio, física de estado sólido (para los sensores y muchos otros componentes del sistema), etc.

2.4 Problemas de integración

Dejando un poco de lado los problemas puntuales que hemos ido viendo hasta ahora, conviene pararse a reflexionar sobre la naturaleza de un sistema de defensa como el que nos ocupa. Aún siendo realmente formidables, los problemas que plantean las armas o ciertas tecnologías concretas son de orden menor comparados con el problema de conseguir que el sistema funcione como un todo. Un sistema de defensa ha de trabajar con un mínimo de coordinación y más aún en este caso en el que la tarea de defensa se divide en varias fases. Se trata de construir un sistema, no una serie de dispositivos de defensa independientes.

En primer lugar han de coordinarse las diferentes fases de la defensa. Los sensores que han detectado los lanzamientos en su primera fase pasan la información a las armas que actúan en esta fase, éstas disparan y se han de recoger datos sobre los impactos conseguidos, qué objetivos han resultado intactos, estimaciones de las rutas, datos sobre objetivos reales y señuelos, etc., si hay tiempo se volverán a reorientar las armas y se volverá a disparar, si no, hay que pasar todos estos datos al control de la segunda fase e informar de qué armas se han dejado disponibles para que se puedan utilizar en fases posteriores y así sucesivamente en cada fase. Y todo esto de forma distribuida porque cada sensor puede actualizar la base de datos de forma dinámica y al mismo tiempo, esa misma base de datos se está consultando para localizar el próximo objetivo. A esto se añade que la base de datos ha de ser redundante y robusta pues daños en un nodo no pueden causar pérdidas de datos.

Algo similar sucede con los sistemas de comunicación, que deben ser capaces de operar en un entorno extremadamente hostil y con unos niveles de radiación y ruido muy elevados. También ha de poder reconfigurarse para asegurar las comunicaciones aunque se produzcan daños en los nodos y poder hacerlo probablemente más de cien veces en pocos minutos. Durante las fases previas a un ataque esto no es tan complejo, en una situación de alarma se puede conseguir que se mantenga un flujo de información adecuado pero en el momento en que empiece el ataque los propios cohetes de lanzamiento destruidos servirán para introducir ruido en el sistema (sobre todo en los sensores) y las cabezas nucleares detonadas en el espacio producirán radiaciones y pulsos electromagnéticos suficientes como para cegar permanentemente a los sensores y cortar las comunicaciones, y eso sin contar con los efectos puramente físicos de una explosión nuclear.

En este entorno se ha de mantener un sistema que funcione coordinadamente tal y como explicábamos antes. Para muchos, la gestión del sistema es la clave de todo el problema de la S.D.I., decidir qué hacer con la información una vez que se ha conseguido. Entre otras cosas se ha de mantener una base de datos con todos los objetivos que existan y la información relevante en cada caso (si es una cabeza nuclear, si es un señuelo, ruta, si ya ha sido desactivado, si está intacto, etc.) ya que no puede repetirse toda la operación de identificación y seguimiento en cada fase y es imposible saber si un misil ha sido ya inutilizado por un láser o si está intacto. También hay que mantener información sobre todos los sistemas de defensa, su situación y estado (cuánta munición queda, qué sensores, computadores y nodos están todavía en funcionamiento, etc.). Coordinar la asignación de armas a los objetivos y mantener la consistencia de la base de datos en todo momento.

El problema de la consistencia es fundamental para conseguir una defensa eficaz pues si una estación con armas resulta destruida sus objetivos han de asignarse a otra estación capaz de alcanzarlos y con suficiente munición como para hacerse cargo de la nueva tarea. También habrá de resolver problemas de datos contradictorios, por ejemplo, un sensor dice que hay 8 misiles y otro dice que hay 12, ¿quién tiene razón?, ¿no es posible que haya 20?, en un sistema de este tipo estas decisiones son cruciales. Y todos estos problemas hay que combinarlos con el hecho de que la base está distribuida y las diferentes copias han de responder de la misma forma pues no puede permitirse que cada copia dé una solución distinta. Por el momento no existen técnicas que aseguren la fiabilidad de sistemas que utilizan bases de datos asíncronas.

Uno de los programas que más frecuentemente se citan como ejemplo de que se puede construir software muy complejo es el que utiliza AT&T para la conmutación en líneas telefónicas (conocido como el Sistema Número 5 de Conmutación Electrónica de AT&T). Este sistema software es del orden del millón de líneas de código, distribuido, en tiempo real, y tolerante a fallos.

Sin embargo este sistema se colapsó en Enero de 1990 durante 9 horas, produciendo unas pérdidas a AT&T que están entre los 60 y 75 millones de dólares. Eso sin contar con las pérdidas producidas en todas las compañías que se basan en el teléfono para su funcionamiento como, por ejemplo, las compañías de reservas de vuelos que estimaron catastróficas sus pérdidas. Este suceso ha demostrado una vez más cómo un fallo en un pequeño programa que forma parte de un sistema mucho mayor tiene unas repercusiones gravísimas en todos los niveles superiores. Lo que es en sí mismo un fallo de un elemento del primer nivel de complejidad, se traduce en un colapso total del sistema (segundo nivel) y una implicaciones importantísimas en el tercer nivel.

El fallo se produjo en un ordenador de Nueva York que "creyó" estar sobrecargado y empezó a rechazar llamadas, otros ordenadores se conectaron automáticamente para hacerse cargo del exceso de demanda, pero empezaron a comportarse como el primer ordenador, creyendo estar sobrecargados y rechazando llamadas. Se formó así una cadena de ordenadores que se iban conectando a los estropeados y que, a su vez, se estropeaban. El fallo parece que se produjo debido a una mejora introducida en el sistema que permitía reducir el tiempo entre el marcado y la señal de conexión, esta mejora introdujo un error en el sistema que se propagó a todos los ordenadores de la red ya que todos tenían instalado básicamente el mismo software.

Evidentemente esta es una prueba más de la vulnerabilidad de los grandes sistemas software aunque también habría que estudiar durante cuanto tiempo ha estado funcionando el sistema a pleno rendimiento hasta que ha fallado y tener en cuenta que el error se produjo al intentar cambiar el programa sin respetar las especificaciones iniciales. [Elmer-Dewitt, en la revista Time, número de Enero de 1990]

El problema de la integración de todos los elementos que componen la S.D.I. ha sido visto frecuentemente como su punto más débil y donde se requieren los avances tecnológicos más importantes para conseguir que el sistema funcione. Muchos de los problemas que se plantean son totalmente nuevos para la tecnología e incluso todavía no se comprenden bien.

2.5 Niveles de complejidad en la S.D.I.

La Complejidad del proyecto de la S.D.I. se puede reinterpretar desde la perspectiva del modelo de tres niveles de complejidad [Sáez Vacas, 1983, referencia a **Marcos Conceptuales**], y quizá sea una de las pocas formas de verlo con la que se pueda abarcar toda la realidad del mismo dada la enorme cantidad de factores relevantes.

Primer nivel de complejidad: Es el nivel de los elementos aislados, en la S.D.I. estos elementos son las armas, las estaciones espaciales, las fuentes de energía, los sensores, sistemas de vigilancia, sistemas de puntería, etc. Todos y cada uno de ellos plantean formidables problemas a la tecnología y a la ciencia, pero no conviene olvidar que son sólo el primer nivel de complejidad. Prueba de ello es que la mayor parte de la controversia no está en este nivel donde se asume que las metas que propone la S.D.I. son alcanzables en un plazo razonable de tiempo. Por otro lado, este nivel es el más maduro de la S.D.I., se conocen gran parte de todos los dispositivos, ya se está experimentando con ellos y se dispone de una base teórica bastante amplia y experiencia suficiente como para confiar en que se conseguirán los objetivos establecidos.

Segundo nivel de complejidad: Del que ya hemos comentado algunas ideas cuando hablábamos de los problemas de integración del sistema de defensa. El segundo nivel de complejidad es la complejidad sistémica, que aparece cuando los elementos aislados (armas, sensores, satélites, estaciones de seguimiento terrestres, radares, etc.) se combinan para formar un sistema, un todo completo que ha de funcionar coordinadamente. La complejidad de este nivel es mucho mayor que la de los elementos aislados y es el problema fundamental de la S.D.I. Es en este nivel donde se han producido las críticas más fuertes, argumentando que si bien los elementos de primer nivel son factibles, la coordinación de todos ellos es imposible dado el tamaño, entorno y las condiciones en las que ha de sobrevivir el sistema.

Tercer nivel de complejidad: Que tiene su origen en la interacción entre el sistema tecnológico y la sociedad. En la S.D.I. este nivel de complejidad juega un papel fundamental, por todas las consideraciones políticas, sociales, económicas y éticas que plantea. Gran parte de las críticas a la S.D.I. se hacen también en este nivel, dejando de lado el aspecto puramente técnico, mostrándose así una vez más la importancia del factor humano y social en la tecnología. Como ejemplo de cuestión política están los tratados internacionales que prohíben la construcción de muchas de las armas que propone la S.D.I.; como problema social está el hecho de que el escudo de la S.D.I. no va a ser perfecto y detrás va a haber toda una sociedad que cree estar defendida; como problema económico basta citar el presupuesto inicial del proyecto, 26.000 millones de dólares, y el coste total, estimado entre varios cientos de miles de millones y un billón de dólares.

Cualquiera que sea el tema que se toque referente a la S.D.I. es conveniente no perder de vista este mapa que nos proporciona el modelo de tres niveles. La complejidad no está sólo en la tecnología, sino también en nuestra capacidad para comprender el problema globalmente y disponer de alguna referencia que nos permita un ordenamiento coherente de todas estas ideas.

3. Software: la piedra angular del proyecto

Gran parte de lo que hemos tratado hasta ahora está relacionado con diferentes tecnologías, láser, física de partículas, satélites, balística, radar, sensores infrarrojos, etc. pero todavía no hemos mencionado la más importante, el software. En este apartado vamos a desarrollar alguna de las características propuestas para el sistema de la S.D.I. y podremos comprobar cómo todo se orquesta alrededor del software. La mayor parte de las críticas y las controversias, de carácter técnico, relacionadas con la S.D.I. tienen que ver con el software. A pesar de que el término "ingeniería software" es ampliamente utilizado, hay todavía muchos que opinan que el diseño y desarrollo de software tiene más de artesanía que de ciencia o disciplina de la ingeniería. La falta de unos principios generales básicos, universalmente aceptados y válidos es una de las primeras barreras con las que se enfrenta todo desarrollo software y más si es de la envergadura del necesario para construir el sistema que propugna la S.D.I.

3.1 Especificación del proyecto

La importancia del software en un proyecto como el de la S.D.I. es fundamental. Generalmente se tiende a prestar más atención a los aspectos espectaculares de la S.D.I. (armas en el espacio, proyectiles de energía cinética, rayos láser, etc.) que a los ordenadores, las comunicaciones y el software que controla todo el sistema. Ya hemos hablado de los problemas que plantean las armas, su puesta en el espacio, la energía necesaria y de la complejidad de conseguir sensores que cumplan los requisitos exigidos. Pero, a pesar de ser problemas formidables hay uno mayor, cómo controlar todas esas armas, sensores, satélites, estaciones terrenas, etc., para conseguir que el conjunto se comporte como se desea.

Problema: *Coordinación de un sistema formado por un número muy elevado de elementos y que ha de sobrevivir en un entorno altamente hostil.*

Solución: *Un programa software que controle y coordine el sistema de defensa.*

Inconvenientes: *La enorme complejidad de un programa con varias decenas de millones de líneas.*

Ese control estará a cargo de un sistema software de entre 10 y 30 millones de líneas de código fuente, ejecutado a razón de 100 millones de operaciones por segundo, capaz de sobrevivir al menos durante 10 años de forma autónoma y sin fallos generalizados y con un tiempo medio entre fallos de dos años. Este es el verdadero problema de la S.D.I.

A este enorme reto que se le plantea a la ingeniería software hay que añadir la necesidad de desarrollar herramientas para poder escribir el código, analizadores, simuladores dinámicos, baterías de pruebas, especificaciones para el desarrollo y todos los elementos que requiere un desarrollo software de envergadura.

3.2 Arquitectura

Un grupo de trabajo de la S.D.I. ha señalado que si no se puede construir el sistema de ordenadores y el software para controlar los sensores y las armas, entonces las características y situación de las armas y los sensores es una elucubración puramente académica. Hay que prestar mucha más atención a los problemas de la complejidad del software y a las pruebas a realizar.

De acuerdo con estas ideas, se ha recomendado una arquitectura abierta y distribuida. Las razones para que sea distribuida son varias, la vulnerabilidad de la opción centralizada (si falla el centro coordinador falla todo), un sistema de defensa no necesita estar altamente coordinado (que en un sitio caigan misiles es independiente de que estén cayendo en otro y las decisiones a tomar son prácticamente las mismas pero independientes) y un sistema distribuido requiere un software no tan complejo (un sistema centralizado ahorra hardware pero aumenta mucho la complejidad del software para coordinar todos los elementos y se reducen las posibilidades de hacer pruebas, pues éstas habría que hacerlas con todo el sistema). Por otro lado, una arquitectura distribuida necesita que se realicen más disparos, un 20 % más, para destruir la misma cantidad de misiles que un sistema centralizado.

Problema: Construir un sistema poco vulnerable y que reduzca de alguna forma la complejidad del software necesario.

Solución: Optar por un sistema distribuido.

Inconvenientes: La solución distribuida es menos fiable en cuanto a su comportamiento y aumenta en mucho el problema de las comunicaciones.

El que el sistema sea abierto es una necesidad, dado que se irán desarrollando diferentes modelos de sensores, armas, comunicaciones, computadores, nuevos nodos, etc. que habrá que ir integrando en el sistema. Un sistema abierto que permita la inserción e integración de elementos inesperados y modificados.

3.3 Desarrollo

En cuanto a la forma de desarrollar el software, se ha descartado el modelo en cascada por su falta de realismo y por la necesidad de volver continuamente a las fases iniciales del proyecto. Parece que se prefiere desarrollar varios prototipos de sistemas software de control en diferentes organizaciones, prototipos realizados con un esfuerzo relativamente pequeño (25 hombres/año).

***Problema:* Diseñar y desarrollar un sistema de gran envergadura que funcione a la primera y hacerlo dentro de unos plazos lógicos.**

***Solución:* Desarrollo siguiendo la metodología de prototipos para reducir al máximo los errores en el software.**

***Inconvenientes:* El alto coste que supone la construcción de varios prototipos del mismo programa.**

Estos prototipos irán creándose de acuerdo con abstracciones y aproximaciones simuladas de los sensores y de las características de las armas. Luego se irán ampliando para substituir las aproximaciones por elementos reales y, a medida que el sistema crece en tamaño y capacidad, se aumentará el nivel de realismo y detalle en las simulaciones.

3.4 Potencia de computación

Una consecuencia directa de tener tan enorme cantidad de código y los requisitos en cuanto a velocidad de ejecución es que el software se habrá de ejecutar, de acuerdo con la capacidad actual, en supercomputadores como el Cray X-MP. Un problema adicional es que estas prestaciones sólo se pueden alcanzar, de momento, con el ordenador en tierra, los ordenadores que se llevan al espacio dan prestaciones un orden de magnitud más bajas debido a las condiciones del entorno en el que trabajan.

Para conseguir el rendimiento apropiado se han de utilizar múltiples ordenadores o diseñar máquinas mucho más rápidas a través de avances en el hardware. Se espera que esto sea posible y siga la tendencia de la tecnología a reducir el tamaño, peso y consumo de los ordenadores y a aumentar su velocidad.

Problema: *Ejecución masiva de software con velocidades del orden de 100 millones de operaciones por segundo, esta velocidad puede ser del orden de varios gigaflops para los sensores.*

Solución: *Nuevas arquitecturas de ordenadores. Computadores ópticos.*

Inconvenientes: *Las nuevas arquitecturas, por el momento, aumentan en gran medida la complejidad.*

Un aspecto importante es utilizar esta potencia para simplificar las tareas, no hacer el software más complejo para compensar la falta de hardware. Además, un incremento en la potencia de los ordenadores puede permitir construir depuradores y simuladores que permitan mejorar en gran medida la calidad del software producido.

3.5 Pruebas

Otro gran caballo de batalla de la S.D.I. son las pruebas. El sistema no podrá probarse por completo hasta que sea el momento de utilizarlo y para entonces no puede haber fallos. La validez de un sistema de defensa radica en la confianza que se deposite en él y por tanto, dada la dificultad para hacer pruebas, hay que elegir la arquitectura y los medios de desarrollo adecuados para facilitar al máximo esta tarea.

Problema: *Se trata de un sistema que, por su naturaleza, no va a poder probarse hasta el momento en que entre en funcionamiento, momento en el cual no habrá tiempo para correcciones.*

Solución: *Construcción modular del software para asegurar la fiabilidad de todas las partes, prototipos para disminuir el tipo de errores, investigación en metodologías de simulación.*

Inconvenientes: *La bondad de las partes no garantiza que el todo funcione correctamente, las herramientas de simulación suelen costar varias veces más que el propio sistema que se construye.*

Un primer paso es elegir un sistema distribuido, por las razones ya comentadas. El tener elementos relativamente independientes evita que haya problemas de coordinación y facilita que se puedan comprobar las partes por separado. La simulación también jugará un papel importante para poder comprobar el sistema bajo diferentes condiciones. Y también será importante realizar pruebas continuas durante toda la vida útil del sistema.

3.6 Tolerancia a fallos

La historia del software muestra que en cualquier sistema hay fallos y más si se trata de uno con la envergadura y complejidad del que nos ocupa. En la S.D.I. los fallos son críticos dada su naturaleza, por eso es crucial reducirlos al mínimo y diseñar todo para tolerar fallos. En lo que atañe al hardware tolerante a fallos hay ya una amplia experiencia y es un campo bastante conocido donde se han hecho avances importantes, sin embargo, el software tolerante a fallos es un campo desconocido donde se ha avanzado muy poco.

La complejidad de tolerar fallos tiene mucho que ver con algunos de los puntos que hemos ido mencionando. El más importante es el hecho de que las pruebas no pueden ser completas. Muchos programas se construyen por el método de prueba y error, mejorando sus prestaciones después de cada intento.

Problema: Hay que asumir que habrá errores, además el sistema de defensa puede ser atacado y quedar inutilizado alguno de sus componentes.

Solución: Construcción de sistemas que soporten fallos, utilizar redundancia, duplicar los sistemas.

Inconvenientes: La tolerancia a fallos del software es muy poco conocida, sólo se pueden tolerar fallos que se han previsto con antelación y no se conoce perfectamente el sistema.

Esta metodología no puede utilizarse en la S.D.I. pues el programa ha de diseñarse para estar libre de errores desde un principio, una práctica inexistente, pues generalmente se confía en una fase de pruebas y de operación experimental en la que los errores se van depurando.

Para solucionar este problema, se ha pensado en la arquitectura abierta y distribuida y en el desarrollo de varios prototipos independientes, ya que la probabilidad de que distintos equipos de programadores cometan el mismo error parece remota. La integración de los diferentes prototipos permitirá reducir la tasa de errores en el programa. También se pretende mejorar las técnicas de tolerancia a fallos a través de sistemas redundantes y producir herramientas más perfeccionadas de análisis y depuración.

3.7 Investigación software

Todos estos planteamientos requieren un esfuerzo importante de investigación y mejora en el campo de la ingeniería software. Muchas de las propuestas realizadas para solucionar los problemas que plantea la S.D.I. son de técnicas y metodologías que se encuentran en su estado inicial y lejos de ser seriamente efectivas. Además de investigar en los problemas que plantea la computación masiva, herramientas de simulación, fiabilidad y tolerancia a fallos, se han señalado seis áreas principales de investigación:

1. Técnicas de verificación matemática: Para la verificación formal de programas, al menos de módulos relativamente pequeños y garantizar así su fiabilidad.
2. Lenguajes de especificación: Que permitan reducir los errores, ya que se obliga al programador a seguir una especificación estricta que asegura la coherencia semántica a medida que se desciende en el nivel de detalle de la programación.
3. Computación paralela, concurrente y distribuida: Dadas las características de la S.D.I. gran parte de los problemas van a estar en estas tres líneas. Actualmente existen bastantes sistemas que explotan las ventajas de estas arquitecturas pero la capacidad para utilizarlos está todavía poco clara. El paralelismo, la concurrencia y la distribución de recursos son, por el momento, fuentes adicionales de complejidad.
4. Equipos de desarrollo: Que podríamos denominar desarrollo cooperativo de software, tomando como base las nuevas redes de comunicación y las potentes herramientas de desarrollo que existen. El problema es cómo gestionar estos equipos y encontrar nuevas formas de estructurar el trabajo de programación.
5. Entornos software: La eficiencia de la programación depende en gran medida de la disponibilidad de entornos de alta velocidad y con una amplia gama de herramientas de desarrollo. Hay que profundizar en el desarrollo de entornos software y estudiar cómo afectan al estilo y forma de programar.
6. Mantenimiento: La S.D.I. introduce un problema totalmente nuevo en el campo del mantenimiento. El sistema ha de estar en funcionamiento permanentemente. Las modificaciones en el software son, hoy por hoy, peligrosas y se ha de investigar mucho en este terreno para garantizar los niveles de seguridad que exige la S.D.I.

4. D.L. Parnas y el pesimismo tecnológico

D.L. Parnas formaba parte del Grupo de Organización de la S.D.I., en 1985 presentó su dimisión alegando razones de profesionalidad pues no consideraba ético que le pagaran por algo que él consideraba que no se podía hacer. Ese mismo año publicó un artículo en "American Scientist" titulado "Aspectos Software en los Sistemas de Defensa Estratégica", en el que enumeraba todas las razones por las que creía que no se podría construir un sistema como el de la S.D.I. Desde entonces se toma este artículo como referencia básica de los que consideran la S.D.I. como un imposible tecnológico.

Los argumentos de Parnas se pueden agrupar en cuatro puntos principales:

Complejidad del Software: El proyecto generará un programa de varios millones de líneas de código, esto significa un número muy elevado de estados posibles en un sistema muy poco regular, la verificación con las herramientas actuales es imposible y no se conseguirán avances importantes en este aspecto en el plazo de tiempo que propone la S.D.I. La tarea de construir un sistema de tal envergadura está más allá de nuestra capacidad como programadores, construir tal sistema requeriría cambiar todos los hábitos convencionales del desarrollo software. La complejidad del software se orquesta alrededor de tres puntos fundamentales: el tamaño del programa, el funcionamiento en tiempo real (el software más complejo que existe) y la concurrencia (paralelismo y distribución de recursos, fuentes de complejidad porque todavía no se dispone de métodos apropiados para programar en estos sistemas). Por otro lado, es un sistema cuyas especificaciones dependen del enemigo, que puede cambiarlas sin previo aviso modificando el entorno de funcionamiento de tal manera que el sistema sea inservible. Por último, se trata de un sistema que no es simplificable, dada su naturaleza no se puede reducir la complejidad permitiendo un mayor número de errores, no hay posibilidad de compromiso, ha de funcionar al cien por cien.

Complejidad del Sistema Soporte: Los elementos a integrar son muy variados y en número muy grande, ya hemos visto que se puede estar hablando de miles de estaciones espaciales. La coordinación será extremadamente difícil en los momentos cruciales y el sistema ha de ser capaz de reconfigurarse automáticamente a gran velocidad sin perder capacidad. La base de datos ha de estar permanentemente actualizada, con datos coherentes y permitir accesos múltiples a gran velocidad. No se tiene experiencia en sistemas de semejante envergadura y con unas constricciones temporales y de tolerancia a fallos tan estrictas y no hay posibilidad de probarlo antes de que entre en funcionamiento.

Complejidad Algorítmica: Han de construirse algoritmos que solucionen un problema del que no se conoce el tamaño con exactitud. Se necesitarán ordenadores de alta velocidad capaces de responder en tiempo real y con una eficiencia

muy alta a un ataque que por el momento es desconocido y que además puede cambiar con el tiempo (por ejemplo, puede que se utilicen modelos antiguos de cohetes como señuelos multiplicando por dos o tres el número de objetivos que se presentan al sistema y no se puede suponer que esos cohetes viejos no contienen cabezas nucleares). A todo esto se añade el problema de escribir especificaciones para un problema que no se conoce exactamente.

Complejidad de gestión: Dado el tamaño del sistema la gestión del mismo es crucial para obtener un funcionamiento adecuado. Las técnicas de gestión de software están poco avanzadas y al no existir ciclo de vida (el programa no se puede probar, las mejoras no se hacen de acuerdo con rendimientos anteriores, etc.) el esfuerzo a realizar para que el sistema sea fiable a la primera será enorme. Parnas asegura que aún teniendo suficientes conocimientos de ingeniería software y disponiendo de todos los recursos que pidiera no podría hacerlo, ni ahora ni en los próximos 20 años.

5. F.P. Brooks, optimismo tecnológico y otras opiniones

A pesar del panorama que describe Parnas no todo el mundo considera imposible construir el sistema de la S.D.I., el ejemplo más característico es F.P. Brooks, un especialista reconocido en ingeniería software, que, llamado a dar su opinión sobre el tema ante una comisión del Senado, afirmó que no veía ninguna razón seria por la que no pudiera construirse el sistema. De hecho, existen sistemas ya construidos que permiten interceptar un misil y que utilizan más de un millón de líneas de código y hasta 30 elementos entre estaciones terrenas, satélites y bases de lanzamiento. Por otro lado, se están haciendo serios esfuerzos por mejorar las técnicas software empleadas en los sistemas de defensa para poder abordar el problema con mayores garantías de éxito.

También se argumenta que existen muchos sistemas que sin haber conseguido la perfección, funcionan correctamente, en el sentido de que son efectivos y fiables. Que un programa contenga errores no quiere decir que no vaya a realizar la tarea para la que está diseñado, en el caso de la S.D.I. un error no es lo mismo que un agujero en el escudo de protección. El grupo del que Parnas formaba parte concluyó sus trabajos en 1985 afirmando que los recursos de computación y el software de gestión del sistema estaban dentro de lo que puede conseguir la tecnología hardware y software en los próximos años. Recientes avances como las investigaciones en ordenadores ópticos (más ligeros y más rápidos que los convencionales), redes neuronales (para paliar la complejidad del software utilizando técnicas de Inteligencia Artificial) y nuevas arquitecturas para proceso de alta velocidad, conceden un cierto margen de optimismo a los que creen que se puede construir un sistema como el propuesto.

Una opinión muy generalizada es que la S.D.I. no debe verse como un sistema perfecto sino como un sistema que haga que cualquier enemigo se lo tenga que pensar antes de lanzar un ataque, al saber que gran parte de él puede ser detenido (y dar tiempo así a un contraataque que no será capaz de parar). Esto es ya una consideración de tercer nivel pero las repercusiones técnicas son importantes. Como Brooks explicó ante la comisión del senado, el problema fundamental es decidir cuál es el grado de perfección que ha de conseguir el sistema (con un rendimiento del 99,9 %, un ataque de 10.000 armas conseguirá que pasen 10 bombas nucleares). El grupo de Estudios Tecnológicos para la Defensa cree que se puede construir un sistema con una efectividad entre el 90 y el 99 % para mediados de la década de los 90.

Una afirmación que también tienen en cuenta muchos es que un sistema defensivo al menos garantiza que un ataque no será tan destructivo, mientras que la construcción de sistemas ofensivos (como los que se han estado construyendo hasta ahora) lo único que garantiza es la destrucción total del atacante.

Hoy por hoy se han alcanzado algunos de los objetivos que se habían planteado pero la gran mayoría de ellos dentro de lo que hemos denominado primer nivel de complejidad. Así, por ejemplo, se han hecho experimentos con éxito en intercepción de misiles. Pero la mayor parte de la crítica sigue centrándose en el software de control de todo el sistema, sobre todo a la vista de los recortes presupuestarios que han afectado especialmente a esta parte del programa. Parece extenderse la idea de que los avances en ingeniería del software son demasiado lentos como para poder abordar los problemas que plantea la S.D.I. en un futuro cercano y es casi una verdad totalmente aceptada que es muy alta la probabilidad de que se produzca un fallo catastrófico debido al software en la primera batalla real. Las líneas de investigación que se consideran más prometedoras son una muestra palpable de ello: sensores a base de superconductores a baja temperatura, técnicas de fabricación de compuestos con mayor resistencia a los impactos, reconocimiento de imágenes no convencionales, interferometría de amplitud y resolución en el nivel de subpixel, microondas de alta potencia, etc.

6. Factores de tercer nivel

Además de todos los que hemos estado viendo y comentando, existen muchos factores de primordial importancia para el desarrollo de la S.D.I. que están dentro de la categoría que denominamos tercer nivel de complejidad. Estos factores se escapan del ámbito puramente tecnológico e introducen problemas de carácter muy variado, social, económico, ético, etc. Al igual que sucede con toda la tecnología, no podemos concluir un estudio sobre un proyecto tecnológico sin considerar este tipo de factores, que en muchos casos son la clave de éxito.

El primer factor que se considera en los proyectos tecnológicos, aparte de los puramente técnicos, es el económico. En este punto la S.D.I. no es diferente de ningún otro proyecto. El presupuesto inicial estimado de la S.D.I. para los cinco primeros años fue de 26.000 millones de dólares, el coste total no parece estar claro pero las estimaciones van desde los varios cientos de miles de millones hasta más de un billón de dólares. Uno de los argumentos que preocupan más económicamente es que un proyecto como el de la S.D.I. no es una inversión única, el mantenimiento del sistema, una vez construido y en funcionamiento, puede costar entre 50 y 200 mil millones de dólares al año. Como referencia se puede tomar el coste del bombardero B-1B: 28.200 millones, desde la investigación inicial hasta la construcción de 100 unidades. Otra referencia interesante es el "Mosaic Sensor Project", que comenzó hace más de once años y su objetivo era producir sensores infrarrojos con un coste de 25 céntimos (de dólar) cada unidad y poder utilizarlos masivamente. El coste actual es todavía 3.000 dólares cada sensor.

Con este presupuesto muchos temen que la mayor parte de las investigaciones se desvíen hacia tecnologías más maduras, aunque menos efectivas, de forma que se puedan obtener resultados con mayor seguridad y justificar el gasto ante la opinión pública. Esto ha ocurrido de hecho y de momento se ha abandonado el trabajo en integración -el software de control de todo el sistema- favoreciendo ciertos campos en los que ya había resultados (interceptación de misiles, rayos láser y satélites). Una decisión política (y por tanto de tercer nivel) que puede afectar decisivamente al futuro tecnológico de la S.D.I.

Otros problemas que merecen consideración, pero sobre los que no nos vamos a extender, son los tratados internacionales que viola el proyecto de la S.D.I., los problemas de seguridad (demasiados participantes para asegurar la confidencialidad de las investigaciones) y problemas de transferencia tecnológica (dificultad para emplear en la industria los descubrimientos y avances que se hagan).

Pero, sin duda, uno de los factores de tercer nivel que más han influido en el futuro de la S.D.I. es la actual situación política mundial. Siete años después de ser propuesto, la caída del muro de Berlín, el derrumbamiento hasta el momento pacífico del bloque del Este, el fin de la guerra fría y de muchas de las tensiones internacionales y la fuerte tendencia al desarme de las dos superpotencias han cuestionado un proyecto como el de la S.D.I.

Es justo mencionar que también ha habido algunos problemas tecnológicos que no parecían abordables y que condujeron a reducir el presupuesto y centrarse en objetivos más realistas. Aunque en opinión de algunos ésta fue la puntilla final a la S.D.I., ya que si los verdaderos y grandes problemas estaban en el software y la integración de todo el sistema sería ahí donde habría que hacer el mayor esfuerzo de inversión y, en cambio, se ha preferido favorecer las investigaciones más específicas en armamento y más relacionadas con la física (recordar lo que se decía sobre la importancia de conseguir algún resultado inmediato).

Y aunque la desaparición del proyecto de la S.D.I. puede dar la razón a algunos de los críticos, tampoco es tan fácil que desaparezca. Y aquí es donde entran en juego en toda su relevancia los factores de tercer nivel. Ya mencionamos anteriormente, en un cuadro, la importancia que tiene en la investigación el presupuesto de defensa de los EEUU. Suprimir radicalmente esta fuente de financiación de las Universidades puede provocar muchos y graves problemas. Prueba de ello es el siguiente cuadro sobre la financiación de las universidades en EEUU:

Universidad	Proyectos de Defensa	Proyectos de la Industria
Johns Hopkins U.	357,4	7,8
Georgia Institute of Tech.	52,1	23,6
M.I.T.	45,8	35,1
Carnegie Mellon U.	43,0	16,1
Stanford U.	40,8	11,0
Austin, Texas U.	37,0	3,2
Southern Ca., L.A. U.	35,1	10,0
San Diego, California U.	23,7	5,9
Urbana-Champaign, Illinois U'	18,6	11,4
Datos de 1987, en millones de dólares USA [IEEE Spectrum, Noviembre 1989, p. 61]		

Otro tema de candente actualidad es que el fin de los proyectos de defensa significa que muchas grandes compañías tendrán que reducir plantilla en todos los niveles de la organización: Hughes Aircraft Co. despedirá 6.000 trabajadores, Lockheed Corporation ha reducido su plantilla en más de 8.000 trabajadores, Grumman Corporation suprimirá unos 3.100 puestos de trabajo, Northrop Corporation entre 2.500 y 3.000, Rockwell International aproximadamente 4.000, TRW unos 1.000, Unisys cerca de 2.000, General Electric 4.000, etc.

Las cifras son suficientemente expresivas como para que cualquiera se dé cuenta del problema que conlleva reducir los presupuestos de defensa cuando la universidad y la industria del país dependen en gran medida de ellos. Ahora quizá se puede entender mejor a los que proponen que el problema de que la S.D.I.

funcione o no es secundario, que lo importante es tener fondos para investigar y que al proyecto se le llame como se quiera. También es importante resaltar que gran parte de esos puestos de trabajo son de personal administrativo y de baja cualificación, lo que dificulta volver a encontrar empleo.

En lo que a los ingenieros atañe, parece que no es tan sencillo cambiar de proyectos de defensa a proyectos comerciales normales. La diferencia de planteamiento entre los dos casos es muchas veces muy grande y básica e impide que un ingeniero de defensa se pueda adaptar a la industria comercial. Por ejemplo, en defensa el presupuesto no es tan importante como las prestaciones, en la industria el presupuesto es vital; la producción en defensa es de pocas unidades con tendencia a sobrediseñarlas, en la industria comercial la producción ha de ser masiva y de productos muy ajustados; la tecnología usada en defensa es tecnología punta, mientras que en la industria comercial hay más interés por tecnología estándar y por mantener bajo el coste.

¿Cómo aunar, pues, todos estos problemas con los que veíamos al principio sobre satélites, armas en el espacio y tecnología punta?. Una vez más se pone de manifiesto la importancia del tercer nivel y se prueba que efectivamente es el nivel superior pues son las cuestiones de este nivel las que decidirán el futuro de la S.D.I. al margen de cualquier consideración tecnológica.

7. Conclusiones

Lo que más nos interesa resaltar de todo lo visto en este capítulo es cómo un proyecto de la envergadura e importancia del de la S.D.I. puede servirnos como base para estudiar una aplicación práctica de las ideas de la complejidad.

No hay sistema conocido donde los problemas de integración, jerarquías, complejidad, incertidumbre, sistemas, fiabilidad y coordinación sean tan importantes y se planteen a una escala tan grande. Desde el punto de vista de la tecnología, la S.D.I. es un reto importantísimo porque implica una innovación radical en muchos campos pero también es una incógnita enorme dados los problemas que plantea.

Para entenderla en toda su extensión y significado hemos de recurrir a las herramientas conceptuales que hemos ido viendo en capítulos anteriores. Sólo a través de ellas se puede llegar a percibir la naturaleza de todas las implicaciones que presenta y su complejidad global. A través del modelo de niveles hemos desarrollado un mapa de la complejidad de la S.D.I. que nos ha permitido profundizar en los aspectos tanto tecnológicos como sociales.

Tecnológicamente hablando, la S.D.I. recoge la innovación y el desarrollo de los más diversos campos y pretende integrarlos en un único sistema. Es muy interesante estudiar las demandas que plantea en cuanto a potencia de proceso, bases

de datos dinámicas, software distribuido, tiempo real, tolerancia a fallos, diseño de software a gran escala, etc. y cómo todos estos campos se han de integrar, considerando todas las implicaciones y consecuencias de cada decisión pues el hardware va a condicionar grandemente al software, el software a la tolerancia a fallos, la tolerancia a fallos a las pruebas, las pruebas a la arquitectura elegida, la arquitectura a la fiabilidad y robustez del conjunto, etc.

Desde el punto de vista de la complejidad, la S.D.I. plantea muchos de los interrogantes que se pretenden resolver a través del estudio de la complejidad. La existencia de varios niveles de diseño, la coordinación de los diferentes sistemas, el gran número de elementos, la redundancia necesaria, la incertidumbre del diseño o la complejidad de la simulación son conceptos que hemos tratado profusamente en la primera parte de estos apuntes y que aquí se pueden ver en un ejemplo práctico concreto como problemas reales y directamente relacionados con la tecnología.

Y para terminar, recordemos lo que se decía en la introducción, instando de nuevo al lector a que sobre las ideas que se le dan aquí sobre tecnología y un proyecto concreto, intente aplicar las herramientas conceptuales que se han ido desarrollando a lo largo de todos estos apuntes y saque sus propias conclusiones sobre su aplicabilidad y validez.

Capítulo 17: An Integrated Framework for Office Information Systems Design and Management

Por Fernando Sáez Vacas y Gustavo Alonso García

Éste es el texto de un artículo enviado para su publicación a una revista internacional.

ABSTRACT

The complexity of Information Technologies is nothing compared with the one that arises when technology interacts with society. Office Automation has been traditionally considered as a technical field, but there is no way to find technical solutions when the problems are primarily social.

In particular, we need a better understanding between the managerial and technical world, offering a coherent, complete and integrated perspective of both. This is the basis for our model, developed as an unfolding of the complexity found in Information Technologies and a matching of these complexities with several levels considered within the Office, Office Automation and Human Factors dimensions. Each one of these domains is studied through a set of distinctions that create a new and powerful understanding of its reality. Using this model we build up a map of Office Automation to be used not only by managers but also by technicians because the primary advantage of such a framework is that it allows a comprehensive evaluation of technology without requiring extensive technical knowledge. Thus, the model can be seen as a principle for design and diagnosis of Office Automation and as a common reference for managers and specialists, avoiding the severe limitations arising from the language used by the last.

1. Introduction

Office Automation (OA) has become a very important application area of technology. We also would like to emphasize the key role the office plays in the development, evolution, competitiveness, and economics of an enterprise. The growing complexity of organizational environments; the trend towards an international market; the strength of regional differences; the need for a strong corporate identity; the capabilities required in order to survive in today's economy; these are only some of the factors which make Office Automation a basic strategic tool.

However, OA is still something fuzzy, unclear, and this prevents things from working as they should. Many managers are already aware of this, and it gives them reasons for fearing technology. OA is obviously necessary, at least as a partial solution to many problems in public and private organizations. Why, then, is it so difficult to implement technology in work environments and to achieve overall acceptance of the equipment? These and similar questions run counter to the traditional point of view, technology as the solution to all ills; nonetheless, they are questions frequently posed by managers, and there are no easy answers.

Problems of a social and human nature cannot be solved from a strictly technical point of view. But to adopt a strictly social perspective is also a mistake, since Office Automation is an applied field of Information Technologies. As a field of study, it requires the use of a conceptual framework for technological and organizational design embracing both social and technical aspects and establishing the appropriate relations giving a global perspective. The goal of our study is to develop a conceptual framework for managers that helps both users and producers to work with a common and integrated understanding of OA.

2. Office automation as a set of distinctions

If there is any application that can be considered to be a synthesis of Information Technologies, it is Office Automation. There is no better field for showing the need for a multidisciplinary, non-specialized and generalist approach. The complexity of Office Automation is obvious. And this is the reason why traditional technological and managerial methods alone are not enough.

Our proposal is to deal with Office Automation through a theory constructed as a set of distinctions. We use the concept of distinction in the sense of Winograd and Flores [31], i.e., as mechanisms within the language to create new domains of action and understanding in OA.

Within Office Automation many different distinctions coexist. At the same time, the perspective of each distinction must be related to the whole, in order to provide a richer and deeper picture of reality. The proper management of Office Automation begins with the recognition of its many facets and of the organization where it is going to be implemented.

We think the first step to a proper management of OA begins with the recognition of a general model as the one proposed here which is neither a methodology nor an implementation system, but rather a conceptual framework for action and interpretation.

3. Understand complexity

There are many reasons for considering technology a complex object. Any manager who keeps in touch with technology is aware of the facts: chaotic, ever changing and disorganized products in the marketplace; incompatible equipment; lack of standards; inadequate systems; lack of human resources to deal with that technology; changes in the organization's structure due to new technologies; lack of motivation to use the equipment, etc..

All this makes very difficult to take full advantage of technology. And since technology does not offer any help in solving problems, it is the user -- especially the

manager -- who has to choose an approach that is relevant to the needs. In many cases, technology models the user's needs and not the other way around. The manager lacks the conceptual tools to evaluate technology, its potential and its subjective value, established according to the needs. To fill this gap we propose the first distinction: a hierarchy of levels of complexity.

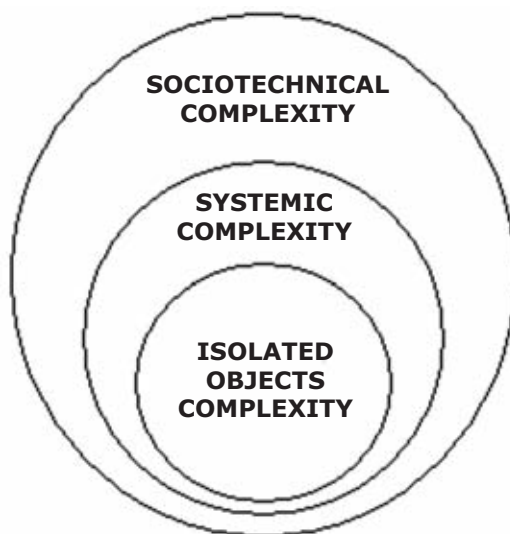


Fig. 1. 3L complexity model

The first level deals with isolated objects. Considered separately, programs and computers, oriented to a narrow application such as text processors, calculators, electronic agendas or electronic sheets, comprise the first level of complexity. As far as the manager is concerned, these are applications with well defined goals.

When those isolated objects are linked together to build a system whose goal is not just text processing, for example, but to give support to some organizational functions, then systemic complexity arises. At this second level, we are dealing with a great number of technical connections and group activities. One example would be a set of computers running first level applications connected through a local network.

The last level stems from the interaction between technological systems and society, the sociotechnical complexity level. It is mainly at this level that the manager works, and it has many characteristics that differ radically from those of the others. Here, all the fuzziness, lack of definition, incongruence and irrationality introduced by human factors appears. Although strictly sociological in origin, this is the highest level and it includes the other two.

This model, which we call the 3-L (three level) model of complexity, was initially proposed by Sáez Vacas in [22]. A brief study of its application to Office Automation can be found in [26] and deeper ones in [1] and [27]. We will use this model as the basic construction for the framework presented in this paper.

3.1 Understand your own office ...

A basic step towards Office Automation is to decide which office tasks are to be automated. Text processing and decision making are certainly completely different activities but both are part of OA. Technology can only be correctly applied if there is a thorough knowledge of what is to be done. One of the initial mistakes of many implementations is to buy technology and then to try to find out what to do with it.

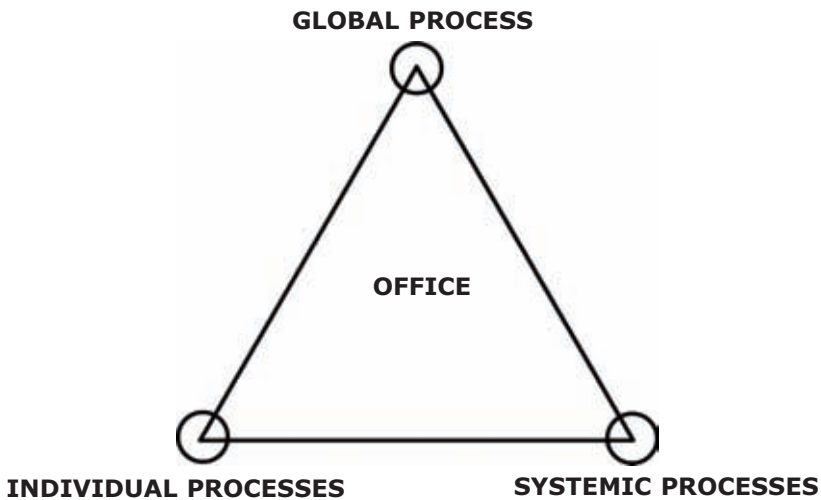


Fig. 2. The office decomposed in three complexity levels

There are many studies analyzing office tasks: reading, writing, making simple calculations, telephoning, mailing, meeting, filing and retrieving of information, etc. Nevertheless, the office is much more. When offices are conceived of only in terms of their manifest behaviour, the above mentioned activities, there is a dissonance between technology and reality, because reality is much richer and has much more variety.

Individual Processes are tasks carried out in an isolated way and without any significant meaning in terms of the organization. No enterprise defines its activity in terms of reading and writing, why, then, does technology do so? Managers must set their goals much higher.

Higher level activities provide meaning and coherence, establishing goals through office functions. These are Systemic Processes, composed of several individual activities, communicating with one another, and connected to build processes within the organization: processing purchase orders, patents and financial reports, etc. Someone not familiar with these environments will just notice first level activities. Managers deal with Systemic Processes.

One step further, all those Systemic Processes must be coordinated to fulfill requirements that affect the organization as a whole. These requirements define higher level activities: the Global Process. This view of the office is our second distinction, depicted in figure 2 and with the same hierarchical meaning of figure 1.

3.2 Apply technology

Only with a thorough understanding of the way each office works can technology be correctly applied. Implemented technology must mirror, as far as possible, the environment's personality. Thus, with the office interpreted as a hierarchy of levels, Office Automation can be seen as having a similar structure. In the past, Office Automation practice has only recognized the lowest level and it is starting to become aware of the second one. Many tools are devoted to individual activities: text processors, electronic sheets, electronic agendas, graphic programs, laser printers, calculators, etc.

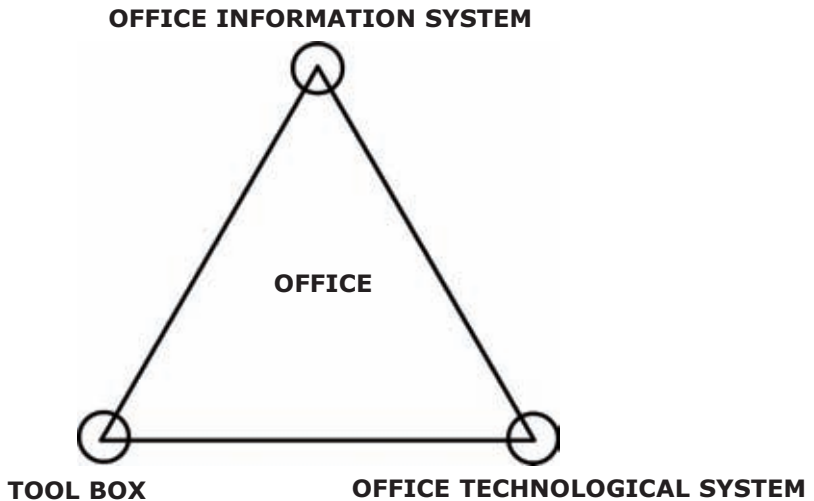


Fig. 3. Office Automation seen through three complexity levels

All this technology is the Tool Box. It deals with very narrow problems and it is devoted to Individual Processes. As a consequence, the user sees technology as a partial solution to unrelated problems. We believe that much of current Office Automation is nothing more than a big tool box.

But Office Automation should be much more than a Tool Box. The next level must offer an Office Technological System in which different activities are grouped together into a meaningful process. There should also be a level in Office Automation for dealing with the Global Process. We call this level Office Information System. Nowadays, technology is far from this not only in technological terms but also in the organization's ability to understand this concept.

3.3 New Distinctions in Information Technologies

To say it in a few words, technology is anything but neutral. It creates an uncontrolled dynamic evolution which, if not channeled, can be a source of real trouble. Very often it goes beyond the users actual needs [14]. To some extent, there is a problem of technology in search of applications, which means an excess of power, low return on investment, lack of adaptation, unjustified difficulty of use and meager benefits from technology.

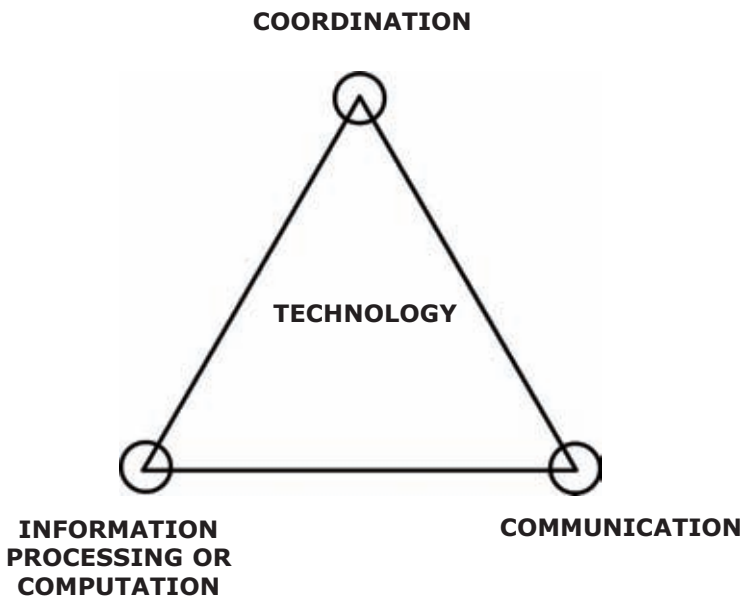


Fig. 4. A new perspective of technology through three complexity levels: the three C's

It is possible to classify technology into three groups and to relate them to the levels in Office and it is possible to classify technology into three groups and to relate them to the levels in Office and Office Automation. Information processing (or computation) is the first level and corresponds to Individual Processes and the Tool Box. This kind of technology can be generally found in Personal Computers and it is oriented towards individual applications.

Communication is the second level. When Individual Processes are combined to form Systemic Processes sufficient communication capabilities are required to integrate individual activities into larger processes. Thus, Communication is the basic component of Office Technological Systems. At this level, the office is a communication problem. Local Area Networks and Micro-Mainframe Links are examples of this type of technology.

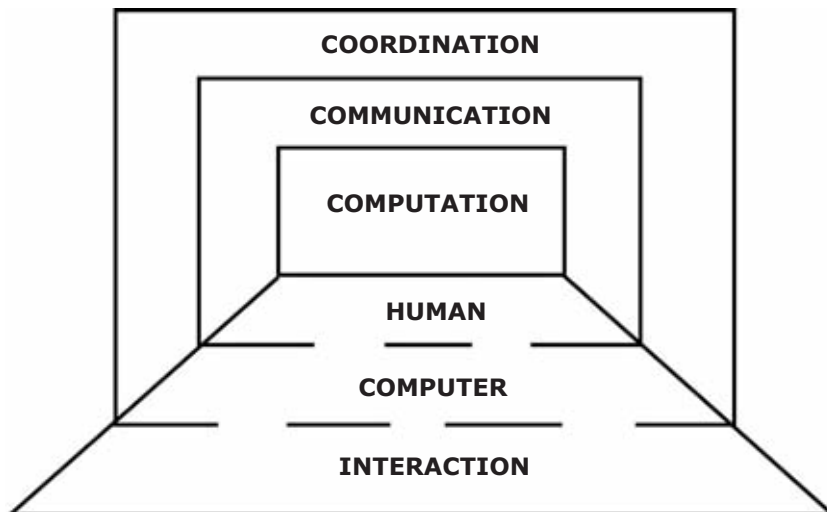


Fig. 5. A new and socially necessary technology: Conviviality, the fourth C

Coordination belongs to the third level. Its function is to support the Global Process. The Office Information System must provide coordination facilities for orchestrating the two lower levels. Some products are starting to appear at this level or, at least, in the border between communication and coordination technologies: Coordinator, Information Lens, Chaos, etc. [27].

Humanization/Conviviality Technology -also known as Computer Human Interaction- deals with the human factor and with sociotechnical complexity. It is a fourth type of technology that makes the other three types usable and then viable.

From a more general point of view, we can consider technology as five types of abstract information processors: T (Information changing with Time), F (Format) and S (Space) [15, 27], M (Meaning) and H processors (Human processing). As we see it, Man-Machine interfaces work as H processors. Figure 6 illustrates this.

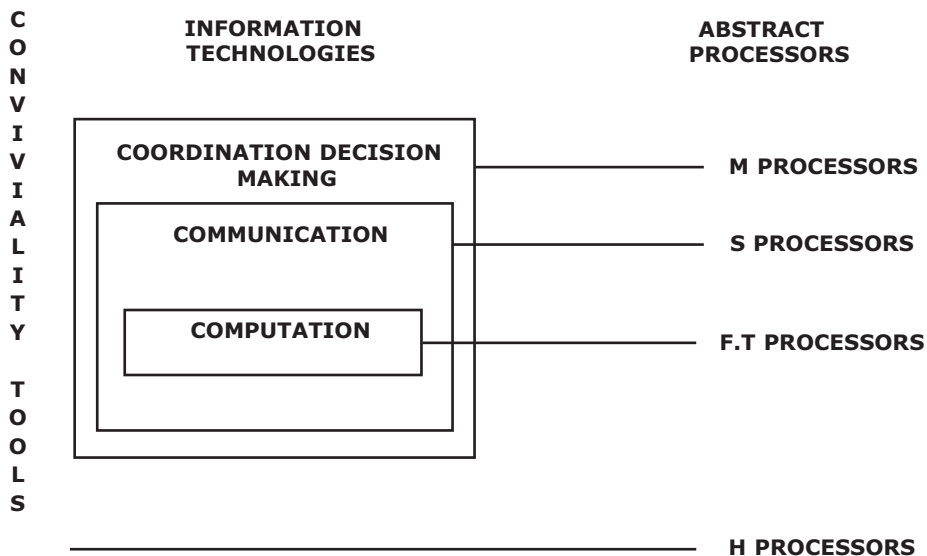


Fig. 6. Abstract processors related to Information Technologies levels

4. The office as a system?

Currently, Office Automation ranks as one of the better resources for enhancing competitiveness. Technological innovation in businesses, in large part, based on Office Automation. But while the technical side is vital, computers are only facilitators: that networks carry meaningful messages and not just bit strings or that memories contain useful information can be seen as one of the key responsibilities of the manager.

There are no methods or methodologies for designing Office Automation. The proposed distinctions, ranging from complexity, office structure, and technology applications to technology, are intended to be the starting point of a conceptual framework used as a general principle for design and diagnosis of Office

Automation. The goal is to give the managers a tool for bridging the gap between their interests and knowledge on one hand and pure technology on the other, providing them with capabilities for evaluating, comparing and choosing the right solution. In order to achieve these goals, these dimensions or distinctions have to be considered as a whole and understood in terms of the appropriate links among them.

4.1 From individual activities to Cooperative Work

Our model is shown in figure 7. Human factors, the fourth dimension, will always be placed as the highest vertex. At the base of the figure are the levels considered in Office, Office Automation and Technology.

Every system is composed of simple objects working together. In Office Automation, one must first surpass the Tool Box level in order to be able to deal with problems at the Systemic Processes and Office Technological System levels. Systemic Processes result from the integration of several individual activities (fig. 8). The number of lines written each day or the improvements achieved in the quality of graphics is of little importance. What really counts is in the number of useful reports produced; the number of patents processed; the speed in answering purchase or sales orders; that is, the overall system. This is the goal of operations research.

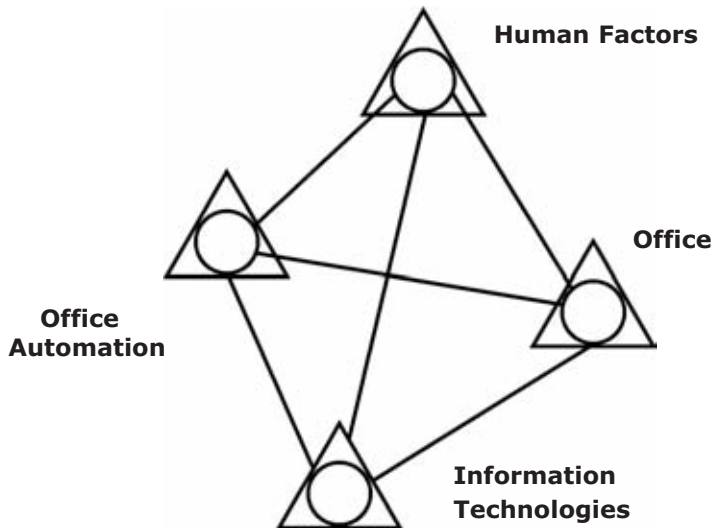


Fig. 7. The whole distinctions set

It becomes obvious that The Tool Box makes individual work easier, but it is not necessarily true that by improving the Tool Box the whole system will thereby improve. This second level can only be improved through the proper technology; through what we call the Office Technological System. The development of Integrated Software was a first step in this direction. Since then, several more applications have appeared which recognize the existence of this second level in the office and which provide methods for connecting individual tasks, addressing the true structure of Systemic Processes: Group Work (figure 9).

The state of the art of computer technology is quite advanced in general but according to our model, it is not so advanced from the office automation and users point of view. There is certain connectivity, as in networks, but: "How do we assure that this connectivity is paying off in productivity? how do we effectively apply these systems to the mission critical activities of the business?".

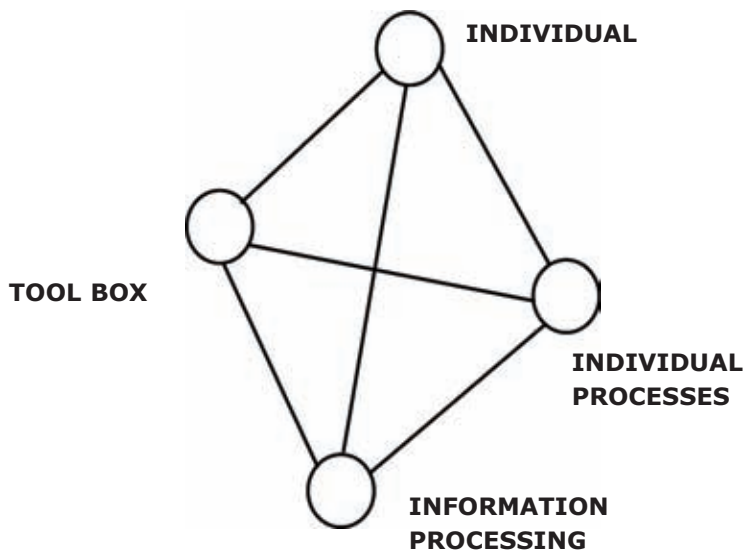


Fig. 8. The individual and his related complexity levels

And there are other authors that agree with our claim: "The personal computer came in to help people do their isolated stand-alone stuff, but the interoperability between them is in very bad shape", Engelbart said, [13]. "80 % of PC's today are involved in task automation, such as wordprocessing, spreadsheets, and bookkeeping ... the return on investment from PC's in task automation is a low 10 to 20 per cent" [Recent report by Nolan, Norton & Co]. "Here has been no measurable increase in productivity in the past decade. That's because personal computers in the Eighties mapped to the old way of working. To achieve the real payoff in the

Nineties, computers have to force a complete reorganization of work", has claimed Apple CEO's Sculley [28].

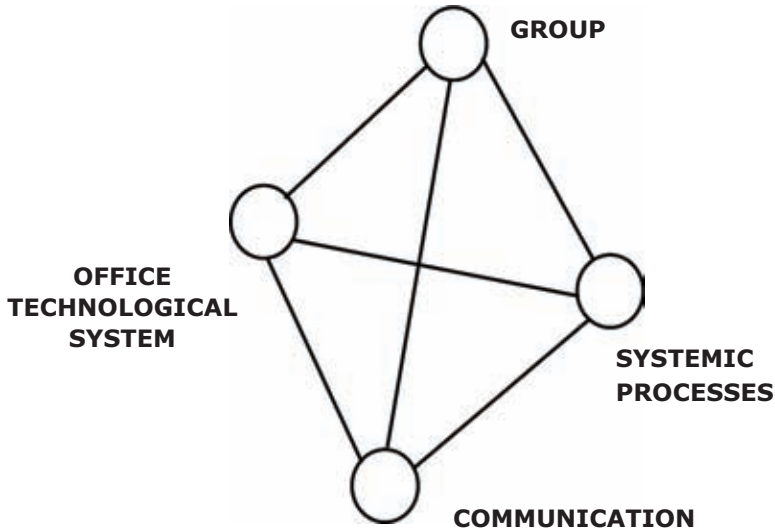


Fig. 9. The group and its related complexity levels

Following these ideas, we see a trend towards giving to systemic processes more importance, a tendency that technology will promptly follow. New terms, such as Business Process Re-Engineering [12,7], Business Process Management or Business Design Technology and Business Process Redesign [4] show the importance of the idea of systemic process. Many of these ideas can be applied in OA. Work-Flow Management Software is one of the first technologies supporting systemic processes [21]. Groupware is on the same line.

4.2 From Cooperative Work to Human Organizations

Our model provides a third level, Global Process, which defines true Office Automation. All functions have a meaning that goes beyond their results or how they work; over and above any other consideration, the office is a Human Activities System [3].

The meaning of Global Process may appear obvious, but it is not easy to apply to Office Automation. Over and above classical measures of performance, there are other considerations more relevant to the manager: changes in organizational structure -generally towards increasing decentralization- retraining of personnel,

reinvestment of time, job satisfaction, acceptance of technology, evolution and growth of equipment along with the organization, etc. Finally, a human organization is a set of systems according to the different organization's images from the different agents [18]. This makes it more difficult to create a sociotechnical system and provides us with an idea of its complexity (figure 10).

All these points are related to the problem of corporate identity. For logical cost reasons, hardware and software are clearly on the path of increasing standardization. Technology thus becomes a very strong factor in the trend towards homogenization, against the equally strong trend towards self identity. In this way, we find a type of technology which, while implemented to improve the organization, ends up producing, as a side effect, a weaker organizational identity, an effect which is intensified when the Tool Box is used as if it were Office Automation.

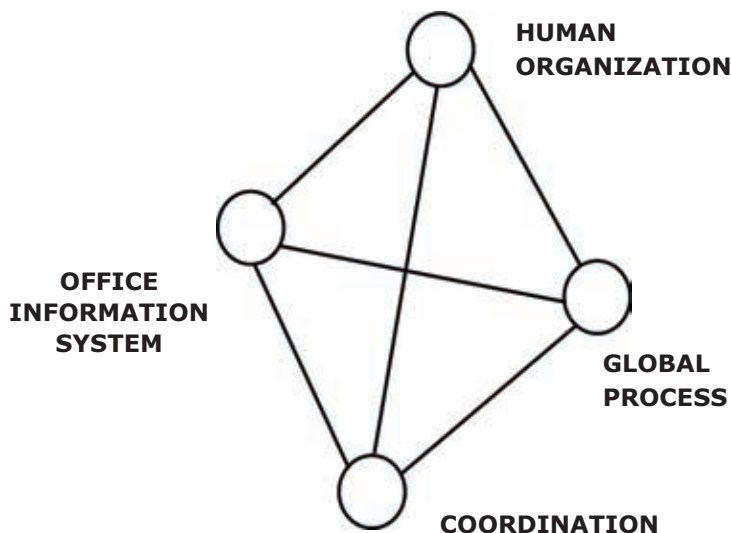


Fig. 10. The Global Process and its related complexity levels

There is some technology at this level. But, more important than that, there is an increase in the number of theories that will surely help to shorten the still long way that lays ahead. The works of Malone [17], Winograd and Flores [31] and Dunham, among others, are valid approaches to the second and third level of our model. Conversation Management and Workflow management will have to converge and produce an integrated theory. There is a fair amount of bibliography in this and related areas, [6] is an example. Our critics to these approaches are based on their failure to incorporate some dimensions of Office Automation (the systems structure, for example, and not only its processes) and Human Factors.

5. The Human Side: Conviviality, as a condition for technological innovation

All the concepts can be synthesized into the diagram shown in figure 11. This provides an ordered image of Office Automation and its related factors. Although the diagram may seem complicated, it is in no way arbitrary. It shows the aligned vertex forming the different distinctions which fall within the domain of the Office, the Office Automation and Information Technologies, with Human and Social Factors, these also include three different levels: individual, group and organization, each placed at the top vertex of its tetrahedron.

The third level of complexity arises with the development 'social factors': a consequence of the complex mesh between humans, office processes and technology in the Office Automation Axis.

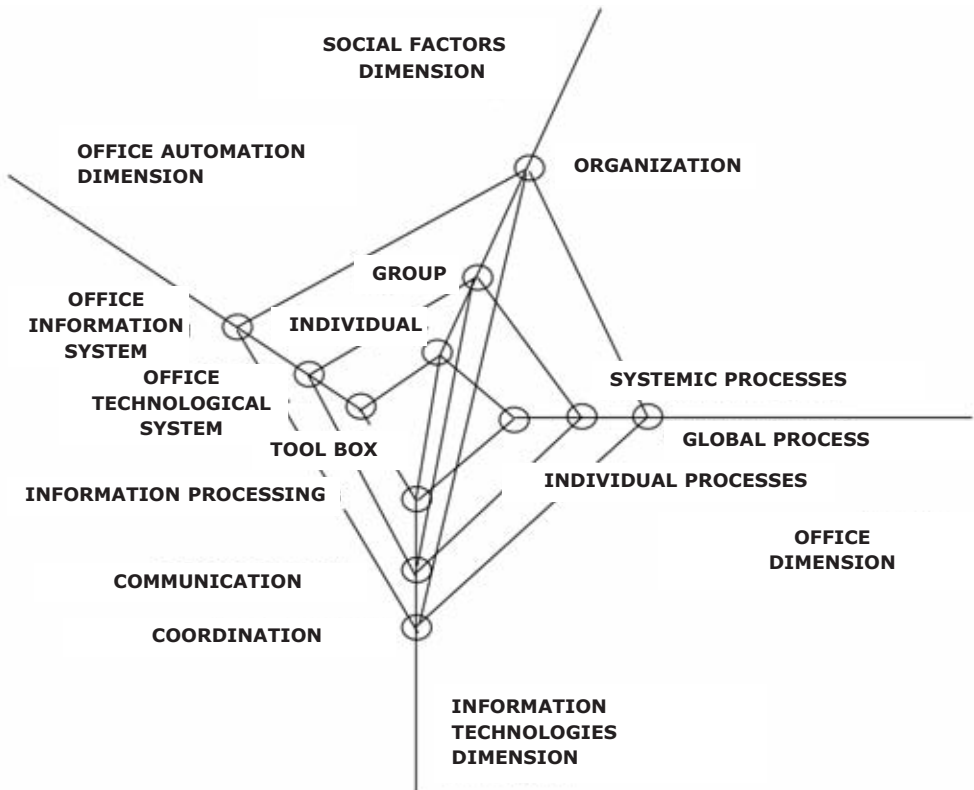


Fig. 11. Hierarchy of complexity levels as a new view of Office Automation

Expanding the personal radius of action can only be achieved if a tool works at the appropriate level. The radius of action of the work group is very different from that of the individual. If individual tools are used within the context of the work group, it will be at the individual level where they will have an impact but will not modify the group's radius of action. It can even happen that, rather than expanding, this radius actually diminishes due to the mismatch between what is expected and what is finally obtained.

Conviviality is the key to technological innovation itself and to the success of technology implementation in work environments. Two aspects of conviviality are user-friendly interfaces and ergonomics but Human Factors play a much richer role when people work in groups and organizations.

The third complexity level is that of sociotechnical complexity, which arises from the interaction between society and technology. It introduces a new domain completely different from the ones traditionally considered. Strassmann [30] states that between 1960 and 1985 approximately 95 % of the references about Office Automation dealt only with its technological side. Today things are different but it is also possible to go too far and put too much emphasis in human factors forgetting technology. Hirschheim's book [8] was an interesting change in focus and a good example of the remaining 5%. This kind of complexity can be seen as intrinsic to "human activity systems" , as Checkland defined them [3].

Global Processes involve different problems which can be grouped together under a new set of subdomains: a) complexity of technology, b) complexity in matching technology with organizations, and c) complexity in matching technology with humans. We represent these three subdomains by the schema of figure 12.

The subdomains affect the manager. Technology has reached an evolutionary stage where it can, and in fact does, deeply transform organizational structures and seriously affect human factors in work environments.

The growing interest in what has been called Human-Computer Interaction proves how important the human factor really is.

Top managers tend to see their organizations from a global perspective, human factors play only a minor role. This point of view, within the traditional organizational structure, would not necessarily have serious consequences but technology provides both individuals and work groups with a degree of power that makes the Human Factors concept very critical. This increased processing and communication power wielded by individuals and work groups represents a major change within the organization.

Psychological resistance on the part of human beings toward changing personal work habits and the fact that many technological tools are still unnecessarily complicated must be overcome. The existence of these barriers requires a thorough study of tool usability, in order to create a truly workable individual-tool interface.

When an organized entity faces/incorporates technology, many important changes result and they must be confronted in order to really reap the benefits of innovation. In the face of this complexity, many possible alternatives can be adopted. Usually, if the manager lacks experience with technology initial prejudice occurs and this may be a major handicap. But managers with at least some or even much experience with technology may also have a problem as well: a failure to take non-technical factors into account.

6. Users and Producers, Managers and Leaders

One of the main problems of technology is the inevitable gap it creates between users and producers. Such a gap is natural when one considers the different approaches toward the use and understanding of technology: to the user it is merely a tool but to the producer it represents a goal. However, such a gap ceases to be natural when it creates a pathological situation, as generally happens in organizations. In Office Automation this problem is exacerbated because there is an even wider gap between specialist and non-specialist. This is one of the main difficulties, found both in business design and information technology, as recently pointed by Keen [12, p.236]: "Today the IT field is at a pivot point. It comes out of a tradition of technocentered thinking, language, and methods and of poor mutual understanding between technical specialists and business managers".

Users are perfectly aware of the environment in which they work; their needs; their particular circumstances; their goals and objectives and their limitations. But, as general rule, they are not able to translate this knowledge into parameters related to technological innovation.

Producers, who create, produce, sell and maintain technology, are familiar with it and its applications, opportunities and advantages. They have worked long enough with technology to, at least, recognize that it poses some problems. But producers do not know details of the work environment and, thus, they can only offer generic products adapted to meet a global demand.



Fig.12. Third level complexity unfolded in three major fields

It could be argued that this gap is not a major problem. Applications such as text processors are general and sufficiently widespread to avoid the need for customization. But text processors are merely a component of the lowest level: the Tool Box. At this level there is enough generality to allow users and producers to coincide in supply and demand.

6.1 Top down / Bottom up

In reality, however, this coincidence is merely an illusion. To the user, what matters is the Global Process, much more than the Individual Process. If the user were capable of translating his needs into the corresponding technology, he would ask for an Office Information System, in the sense we are proposing, or, if not available, Office Technological Systems, not just a Tool Box. To him the hierarchy is top-down, with almost no relevance at the lowest levels.

To the producers it is just the other way round. Due to technological and market constraints, the producer gears his offer towards the Tool Box concept by means of generic applications having no technical or conceptual difficulties, in this way guaranteeing a wider market. Upper levels require a much more refined product; more knowledge about user needs is necessary, and the potential market is much narrower (the upper extreme would be custom design). The hierarchy as the producer perceives it is bottom-up, with Individual Processes as a first goal.

Our model of Office and Office Automation in effect moves these two perspectives closer together. With this model, the user can establish technological needs with specific reference to particular activity, while the producer can recognize the existence of upper levels in the demand and react accordingly.

6.2 Managing complexity

Many methodologies used to implement technology underscore the role played by the leader of technological innovation. The ultimate success of technology and future user satisfaction depend largely on the efforts of this person, who seldom has the resources necessary for achieving these goals. An implementation methodology can offer a more or less effective way to introduce technology in work environments; to plan the organization's evolution, or even take into account more advanced problems such as productivity measures or cost justification. But no methodology can say whether technology is applied at appropriate levels.

Managers interested in technology can read books and articles and find a "solution" to almost any kind of problem and, to a lesser extent, methodologies for implementing those "solutions". But in spite of that, the manager lacks the required reference point but based on needs and according to the characteristics of the organization. With our model, the manager has a very powerful conceptual framework for completing a sociotechnical design, as well a reference model upon which to map it. Managing Office Automation is equivalent to managing third level complexity. In general, technological innovation management is equivalent to complexity management.

Today's managers have become Information Systems Managers [10] and they must think in terms of this new role. Models such as the one proposed here facilitate this task by having a better understanding of the technical and managerial world, offering a coherent, complete and integrated perspective of both. Both, users and producers, can thereby have a shared understanding, with a common language to facilitate actions.

7. Conclusions

We have proposed a conceptual model for Office Information Systems, a model that evolves through three types of complexity to form a hierarchy. The model integrates four domains: information technologies, office processes, social factors and office systems. The result is a new and complete framework for the design and management of O.I.S.

The model has been presented in the form of a group of distinctions, at once powerful and easy to remember. To accentuate its dialectic character we have used as many graphic representations as possible. These figures can be summarized by Figure 11.

In [1] and [27] it has been shown that this conceptual model reflects the various possible perspectives of the office, as synthesized in [8]. In [27] the model has been analyzed as a tool for developing, in conjunction with cybernetics, a new vision of technological innovation in organizations through office automation.

7.1 A non-technical language for managers

The primary advantage of such a framework is that it allows a comprehensive evaluation of technology without requiring extensive technical knowledge. In other words, it provides a new and accurate language for expressing any kind of technological need posed by the organization, while avoiding the severe limitations arising from the language used by specialists.

By using the model, the manager can easily identify in which levels technology is to be applied, and always has at hand the references which make evolution possible; not only with technology but also coherently with corporate objectives. These three levels (Individual Processes, Systemic Processes and Global Process) constitute a step-by-step approach to Office Automation as well as a non-traumatic way for assimilating technology into work environments. At the same time, it permits managers to identify the most appropriate technology for each level, and to be guided by what it is actually done in the office, instead of what technology has to offer.

In this way, managers have a very powerful methodology for planning their technological strategies without having to face solely technical issues. As concerns technology, the model points out a very clear path for innovation and research. Once the first level (The Tool Box) is overcome, technology should concentrate on solving cooperation issues, integrating first level tools into full solutions to Systemic Process problems. This trend has already begun, as shown by the growing interest in Computer Supported Cooperative Work. But the fields where major research is still needed are basically Office Information Systems; the approach of technology to corporate activities; understanding the organization as a whole, and the study of implied factors (social, job satisfaction, organizational changes, corporate goals, evolution, competitiveness, etc.).

We would like to underscore the potential the model shows for serving as a common reference for the two major participants in Office Automation, users and producers, integrating both perspectives, top-down for the first, bottom-up for the latter, and opening new domains for action.

7.2 Opening new perspectives

A very important aspect of our study has been its focus on Office Automation from the point of view of complexity; in fact a very uncommon perspective in the fields of technological research and application. Unfortunately, studies about complexity, from the pioneer works of Simon [29] to the works of Morin [19] and Le Moigne [15] among others, are essentially unknown to managers.

We can feel justifiably hopeful about this last point, if books such as Pagels [20], who recently passed away, can be seen as representing a new trend in the recognition of the complexity issue. In his study, he proclaims the computer to be the basic instrument of the science of complexity: "the great unexplored frontier is complexity" (p. 12); "I am convinced that the nations and people who master the new sciences of complexity will become the economic, cultural, and political superpowers of the next century" (p. 15).

We believe that, nowadays, the sciences of the artificial [29] are being replaced by the sciences of complexity, particularly when they include the human factor, as is underscored by the focus of the first Conference dealing with Critical Issues, which was held in November 1990, by the Association for Computing Machinery (ACM). There were two central subjects to debate: Managing Complexity and Modeling Reality. These two principles have guided our paper.

8. References

- [1] Alonso, G., "Conceptual Framework and Systemic Modeling of Office Automation", Master Thesis (E.T.S.I.Telecomunicación, Universidad Politécnica de Madrid,1989 -in Spanish-).
- [2] Bair, J. H. and L. Mancuso, "The Office System Cycle", Hewlett-Packard, Palo Alto, California, 1985.
- [3] Checkland, P., "Systems Thinking, Systems Practice", J. Wiley & Sons, Chichester, 1981.
- [4] Davenport, T. and J. Short, "The new industrial engineering: Information technology and business process redesign", Sloan Management Review, summer 1990, pp. 11-27.
- [5] Denning, P., "Beyond formalism", American Scientist, vol. 79, 1990, pp. 9-10
- [6] Greenberg, S., "An annotated bibliography of computer supported cooperative work", ACM SIGCHI Bulletin, July 1991, pp. 29-62.

- [7] Hammer, M., "Reengineering work: don't automate, obliterate" Harvard Business Review, July-August 1990, 4, pp. 104-112.
- [8] Hirschheim, R. A., "Office Automation: A Social and organizational Perspective", J. Wiley & Sons, Chichester, 1985.
- [10] Hoffman, G. M., "Every manager is an Information System Manager Now, or, Managing User-Controlled Information Systems", Information & Management, vol. 9, no. 11 (November, 1986), pp. 229-235.
- [11] Illich, I., "Tools for Conviviality", Harper & Row, Nueva York, 1973.
- [12] Keen, P., "Shaping the future. Business design through information technology", Harvard Business School Press, Boston Ma. 1991.
- [13] Kelleher, J., "The personal stuff is great, but ..." in Special Report: "Advances on the PC front", Computerworld, May 20, 1991, p.111.
- [14] Kundig, A. T., "Future Computer and Communication Supported Working Environments", Research into Networks and Distributed Applications, R.Speth Ed., Bruselas, 1988.
- [15] Le Moigne, J. L., "La théorie du système général. Théorie de la modélisation", P.U.F., Paris, 1977, 1984.
- [16] Le Moigne, J. L. and E.H. Sibley, "Information-Organization-Decision: Some strange loops", Information & Management, 1986, 11, pp. 237-244.
- [17] Malone, T. and K. Crowston, "What is coordination theory and how can it help design cooperative work systems?", Proceedings CSCW 90, pp. 357-370.
- [18] Mélése, J., "Approches systémiques des organizations", Ed. Hommes et Techniques, Suresnes, 1979.
- [19] Morin, E., "La Méthode: 1. La Nature de la Nature ", Seuil, Paris, 1977.
- [20] Pagels, H. R., "The dreams of reason. The computer and the rise of the sciences of complexity", Bantam Books, N.Y., 1989.
- [21] Pastore, R., "Beyond the beginners' slope", in Special Report: "Advances on the PC front", Computerworld, May 20, 1991, p.96

- [22] Sáez-Vacas, F., "Facing Informatics via Three Level Complexity Views", presented in the X International Congress on Cybernetics, Namur, Belgium, 1983, pp. 30-40.
- [23] Sáez-Vacas, F., "Some framework ideas for Software Engineering Education", in Proceedings of International Computer Symposium, Taiwan, 12-14 December 1984, vol. 1, pp. 150-156.
- [24] Sáez-Vacas, F., "Conviviality, Complexity, Computers and Informatics", La Vanguardia, 24 February, 1985, (in spanish).
- [25] Sáez-Vacas, F., "Personal Computers: Towards a World of Informatic Machines", Fundesco Ed., Madrid, 1987, (in spanish).
- [26] Sáez-Vacas, F. and G. Alonso, "Proposal of a three level complexity model for office automation", presented at the 33rd Annual Meeting of the International Society for General Systems Research (P.W.J. Ledington (Ed.), Edinburgh, Scotland, 2-7 July 1989), vol. IV, pp. 197-204.
- [27] Sáez-Vacas, F., "Complex Office Automation", Fundesco Ed., Madrid, 1990, (in spanish).
- [28] Schlender, B. R., "The future of the PC", Fortune, August 1991, pp. 40-48.
- [29] Simon, H. A., "The architecture of complexity" in "The sciences of the artificial", The MIT Press, Cambridge, 1969, published in Proceedings of the American Philosophical Society, 106, pp. 467-482, 1962.
- [30] Strassmann, P. A., "Information Payoff. The Transformation of Work in the Electronic Age", Free Press, New York, 1985.
- [31] Winograd, T. and F. Flores, "Understanding computers and cognition. A new foundation for design" Addison-Wesley, Reading, Ma., 3rd printing, 1988.

Capítulo 18: A cybernetic theory about computer interfaces and human factors within a framework of technological innovation

Por Fernando Sáez Vacas

1. Introduction

This paper is based on the following postulates taken from a book recently published by this author (Sáez-Vacas, 1990(1)):

- a) Technological innovation in a company is understood to be the process and set of changes that the company undergoes as a result of a specific type of technology;
- b) The incorporation of technology in the company does not necessarily result in innovation, modernization and progress;
- c) The very words "modernization" and "progress" are completely bereft of any meaning if isolated from the concept of complexity in its broadest sense, including the human factor.

Turning to office technology in specific, the problem of managing office technology for business innovation purposes can be likened to the problem of managing third level complexity, following the guidelines of a three-level complexity model proposed by the author some years ago (Sáez-Vacas, 1983).

Lastly, managing complexity is an application of one of the main principles of Cybernetics, the Law of Requisite Variety (see Ashby (1956), Beer (1985), Mèlèse (1979) and many others).

2. A new theory on office automation

Both Sáez-Vacas (1983) and Flood (1987) incorporate in their respective investigations on complexity the interaction between the organized complexity of artificial systems and the disorganized complexity of man and society (Sáez-Vacas) and the complexity of homo sapiens introduced by human activities (Flood).

Office automation produces artificial systems --office systems-- for which Sáez-Vacas recently developed a complete theory (Sáez-Vacas & Alonso, 1989) (Sáez-Vacas, 1990(1)) based on the afore cited three-level model (1983). The automated office is an anthropotechnical system in which we find three levels of complexity.

To provide the framework for the subject of this paper, below we are furnishing a very brief general summary of this theory.

This theory is synthesized in the diagram shown in Figure 1. The world of offices and technology comprises four dimensions and three hierarchical levels: for example, the three levels of crescent complexity in the office automation dimension are, starting from the bottom, the Tool Box (i.e., various tools such as word processors, electronic sheets, graphic programs, data base packages, etc., devoted to individual activities), the Office Technological System (in which the above tools are, technologically speaking, interconnected and integrated) and the Office Information System.

In this paper we are interested in focusing our attention on the role of the human and social factors, i.e., the fourth dimension, which, in turn, encompasses three different levels: the individual, the group and the organization. In relation to this dimension emerges the word "conviviality."

In the early 1970s, Ivan Illich was the first to coin the term conviviality (Illich, 1973). In his view, there are three requirements a tool must meet to be considered convivial: it must be efficient without diminishing personal autonomy; it must refrain from creating masters and slaves; and it must expand the radius of our personal actions. Thus, convivial is the best term for describing what office automation must be.

The third level of complexity in its broadest sense arises with the development of what are called in Figure 1 Social Factors and is therefore a consequence of the complex meshing of humans, office processes and technology within office automation dimension. This gives rise to the notion of conviviality as a requirement for the evolution of this kind of system (and for meeting the cybernetic law of requisite variety, applied here to complexity). In the above three principles of conviviality no advice is found concerning productivity, applications, organizational structures or management methods. The principles only point out what the relationship should be between the user and the tool, whoever and whatever they may be.

We propose applying these rules, or principles, to all the levels identified in office automation. Office automation, like any other tool, must be at the service of the individual, not the other way around. To achieve this, we normally need to know what we want to do, how it will be done and who is going to do it. Some of these questions have been answered through the distinctions made in Figure 1; this Figure shows us how to structure our knowledge of Office Automation, Technology and Organizations. But this would merely involve a conceptual exercise if we do not try to go further. And for this reason we are introducing a new understanding of these distinctions through the complexity/conviviality tandem.

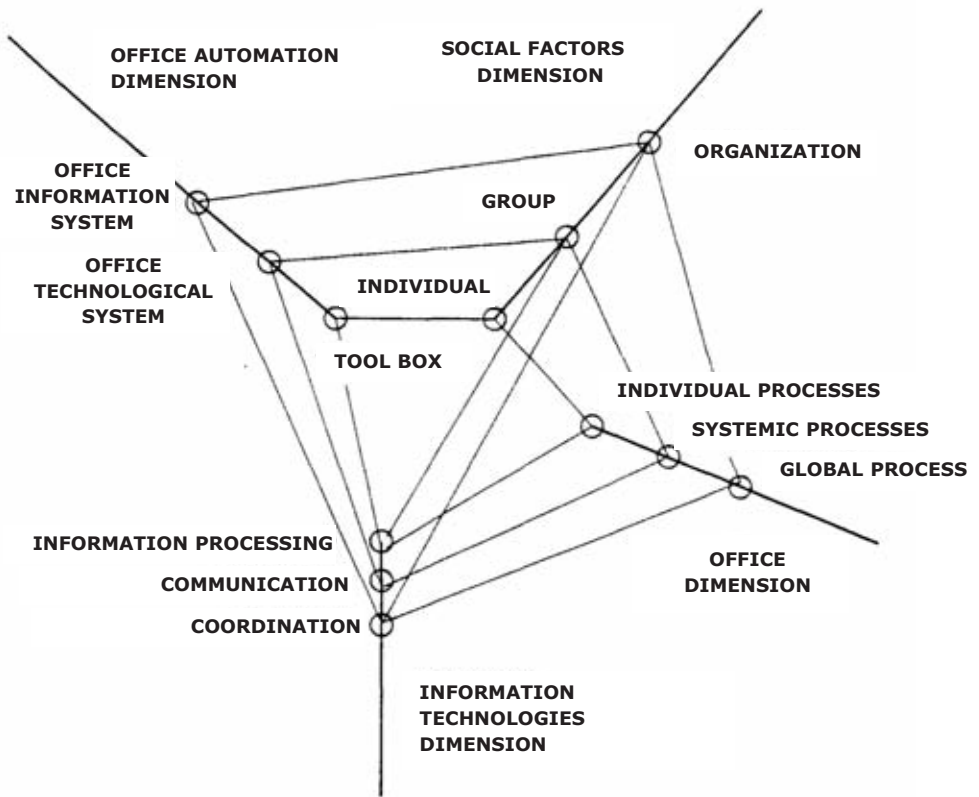


Figure 1

With a hierarchy like the one proposed here, we can ensure that the right balance is established among all levels, at the same time that the levels at which complexity must be considered become easier to identify. The first level is the individual level, which is perhaps the best known, given that most people work at this level, making it the only level that exists in the minds of many. However, not until recently was it recognized that it is more interesting to focus on group work. Not too long ago groups worked with individual tools, and the individual had to struggle with the complexity generated by the lack of balance. This situation disrupted not only the work of the individual but also the work of the group as a whole.

Individual tools used in a group environment do not enhance efficiency; nor do they respect personal autonomy. It is therefore necessary to think in terms of an Office Technological System as the proper tool for group work. For this same reason, the individual tools used by the work group, where the primary need is actually communication, to process information creates an undesirable dependency on the

elements providing that communication. The solution is, again, the Office Technological System, whose main goal, after all, is to support work group communications.

The third rule, that of expanding the personal radius of action, can only be met if the tool works at the appropriate level. The radius of action of the work group is very different from that of the individual. If individual tools are used within the context of the work group, their impact will be felt at the individual (and lowest) level; however, they will not alter the group's radius of action. As a result of a discrepancy between expectations and the final outcome, instead of expanding, this radius may actually diminish.

Conviviality is a key both to technological innovation itself and to the successful implementation of technology in work environments. It is important that human factors/conviviality also be considered an aim of support technology. Two very well-known aspects of conviviality are user-friendly interfaces and ergonomics, both of which are certainly important, though Human Factors play a much fuller role. Even the concept of user friendly must be enlarged to encompass groups and organizations.

3. Office systems are human activity systems

The remainder of this paper will focus on the third complexity level, which is where technological innovation processes unfold. Integrating office technology into a company leads to non-structural problems which are difficult to express if oversimplification is to be avoided. One notable aspect of these problems is that by participating in the system, designers and implementers inevitably become immersed in the very process of change.

This type of systems does not come under systems engineering but belongs to what Checkland (1981, p. 111) calls human activity systems, consisting of a number of activities linked by some principle of coherency. Such systems encompass both those fuzzy types of problems that must be tackled by company managers and many even poorer defined social problems. The nature of their complexity is multidisciplinary and, in the case of office automation, technological and socio-economic; this is why offices systems should be approached as human activity systems. In other words, we are facing a very special type of theoretically systemic problem.

This approach towards office systems, however, has been practically non-existent in research and writings. Strassmann (1985) states that between 1960 and 1985 approximately 95% of the reference works on office automation dealt only with its technical side. Hirschheim's book (1985) and very few others have provided an interesting change in focus and a good example of that remaining 5% of reference works.

4. The basic structure of the third level of complexity

To examine this level of complexity, we use the classical technique of "divide and conquer," hypothetically breaking this level of complexity down into three simultaneously intervening sub-levels (see figure 2) presented below:

- a) The complexity, strictly speaking, of office technology;
- b) The complexity of the relationship between the human organization (the company as an organized system) and office technology;
- c) The complexity of the relationship between the human factor (as individual components of an organization in which jobs are assigned) and office technology.

The three categories of complexity, illustrated in the graph by two ovoids and a circle, represent to us many other types of variables, all of which must be concomitantly controlled when automating a company.

5. The complexity of office technology

The complexity of office technology is viewed one way by the company's managing director, another way by a company clerk and yet another way by an office technology expert, just to cite a few of the agents involved in the innovation process. Nevertheless, from every perspective the complexity of office technology is not only vast but greater than the average complexity (in variety, forms of control) of any organization and its members. This situation is negative for companies because it does not meet the law of variety which states that the variety of the control system --in this case the organizational structure of the company-- must be equal to or greater than the variety of the system being controlled, a role which in this case is played by office technology. Otherwise, the system as a whole is unstable.

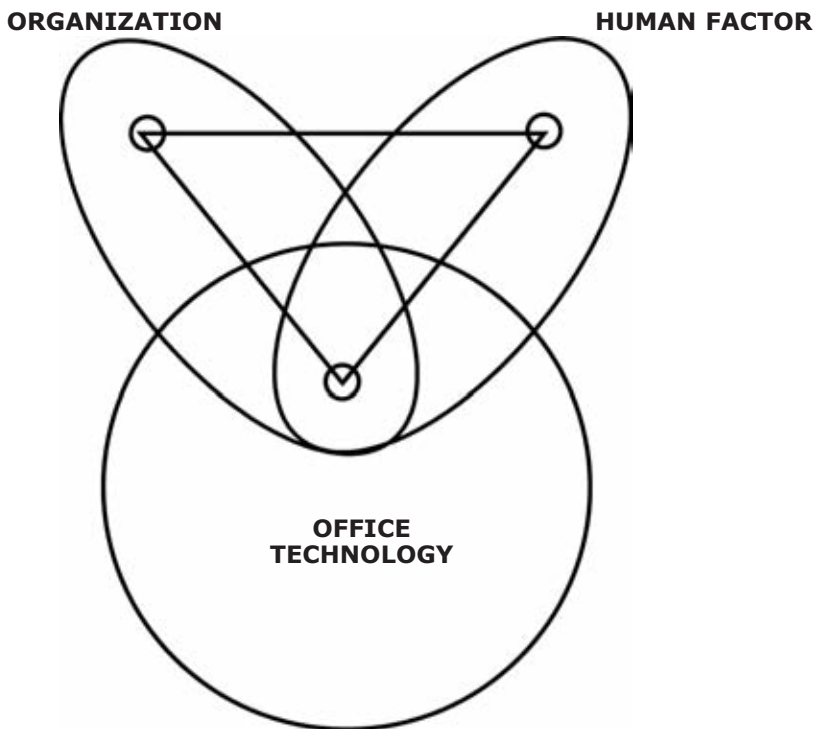


Figure 2

Figure 3 attempts to synthesize the elements which foster office technology complexity and no doubt represent intrinsic common barriers that the current state of technology itself paradoxically raises against technological innovation.

We could analyze each one of these elements, but, to avoid lengthening this paper unnecessarily, we will leave this analysis partly up to the reader's intuition. Various chapters of the afore-cited book (Sáez-Vacas, 1990) address this matter. Here we will simply give a brief overview:

At the level of computation (or Information Processing - see Figure 1), the variety of computing capacities and properties encompasses machines ranging from personal computers of various families and capacities to mainframe computers, including in between all types of mini-computers and an ever-growing assortment of working stations. As far as memories are concerned, there exist all types of technology, be they 5" 1/4 or 3" 1/2 diskettes and recording tape or optical CDs, not to mention all the complementary apparatus. Turning to software, what technician is capable of getting a handle on the confusing array of available programs,

which vary from those that resolve simple, specific office functions to those of the ever-richer group of integrated packages?

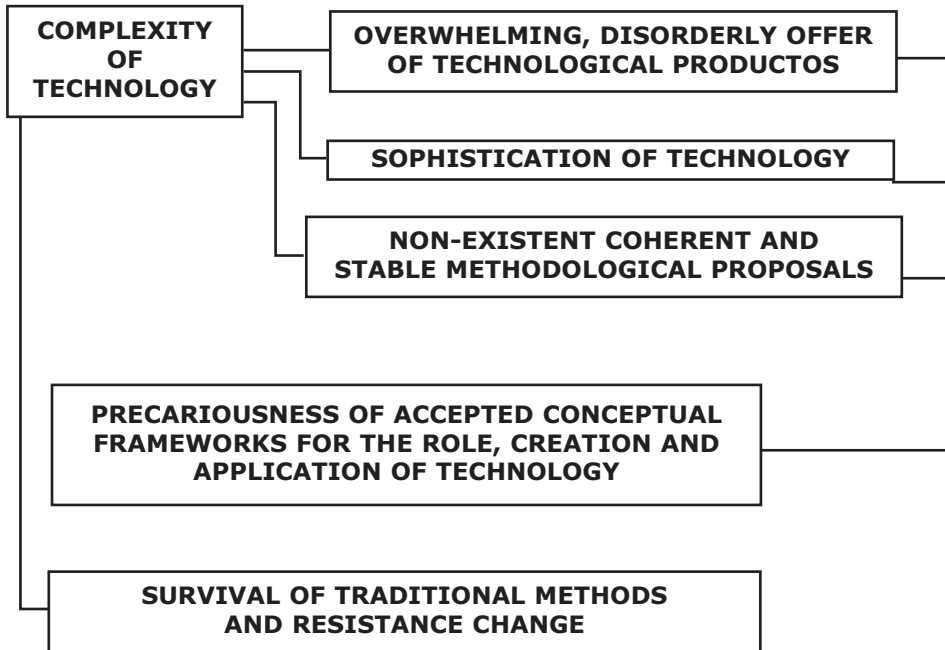


Figure 3

At the level of communication technology, the perspective is equally overwhelming and, to a certain extent, too broad intellectually. We have local networks operating, with highly diverse technical systems, long-distance network connections, new generations of PABX telephone exchanges and already emerging are integrated digital service networks and future IBCN networks.

The complexity of technology does not end with its variety and sophistication. Rather, it is becoming accentuated by the practices of an aggressive, brash market in which the multiplication of companies selling all types of products, tools, machines, services, solutions, etc., is creating a haze of confusion in the minds of customers. The picture is completed by the proven fact that, in general, the methodology for applying and introducing information technology is in evolutionary terms lagging behind said technology by various generations.

The real degree of technological complexity as schematically described above is reflected in the high volume of failures, problems and resistance relating to its introduction, outside of its potential for providing the promised working advantages.

The classical argument that blames the resistance put up by humans or by an organized system for the barriers that are blindly raised to the marvels of technology is only half-true. The other half of the truth is that the excess of technological complexity, that is to say, its unsuited capacity in human terms, elicits conservative reactions and creates a void of ideas about the role, the creation and the application of technology, which fuels the vicious circle depicted in Figure 3.

6. Adapting complexities: an application of the law of requisite variety

After making the above statements, and returning once again to Figure 2, it can be asserted that, as a rule, it is impracticable for the organization (or the human factor, i.e., each user) alone to bring about the change and increase in complexity necessary to rise to a level of technological requisites. The office technology industry must also be asked to meet the company partway by reducing the complexity of its products and services, at least in the area visible to the user. These two complementary strategies are reflected in Figure 4 by means of some symbols typically used by Beer to represent the mechanisms for amplifying and reducing variety (Beer, 1974, 1985), which will be discussed at some length below.

6.1 Organization versus Office Technology

It must be accepted that the organization will always have to increase its complexity if it wants to evolve, and the case we are analyzing is no exception. On the contrary, this is a special case with a superior approach, like the one described by Mélése (1979), among others.

Some of Mélése's ideas can be summarized as follow:

- a) An organization is a system of mental representations that are difficult to separate from the environment;
- b) All companies must address the problem of information from an overall viewpoint;
- c) Autonomy and innovation require complexity;
- d) This complexity must be distributed among all the levels of an organization.

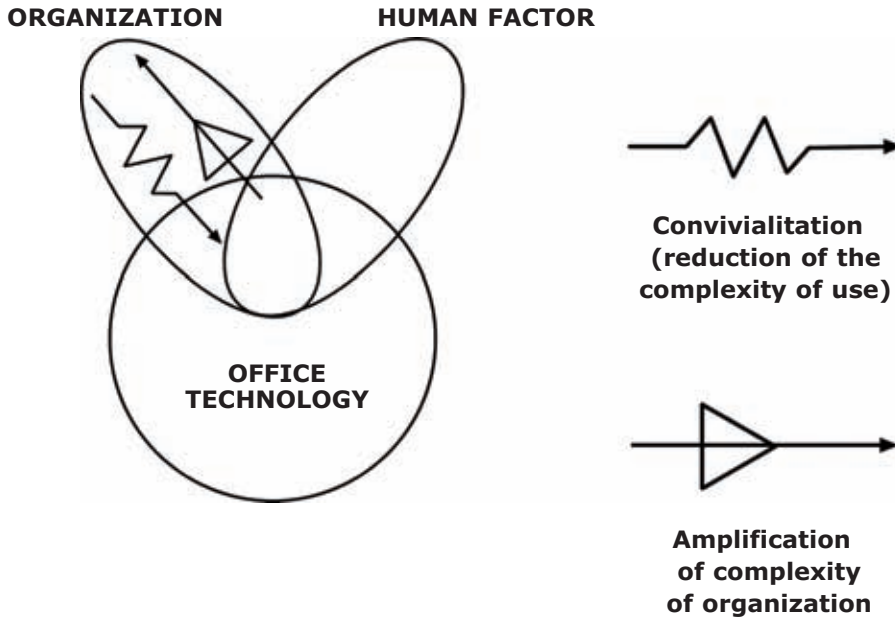


Figure 4

Let us explain. In an organization the effects of numerous "transversal systems and kinds of logic" (technological, commercial, financial, social, trade union, spacial, symbolic, etc., systems) intersect and combine and, for the most part, are determined and controlled outside of the organization, in some part of the environment. In other words, each organization is a node of the larger political-social-economic system, and this node is in turn a system whose interacting components are the mental representations of a variety of external and internal agents.

It is clear that one aspect of the difficulty of the problem resides in that having this node evolve in terms of organization, information and communication in order to follow the evolution of the environment requires equipping it with a variety of ways to perceive, associate and combine numerous dissimilar variables.

Office technology forms part of the external technological system, as well as of the commercial, social, trade union and other systems. Strictly speaking, it is a typically organizational technology, for it sets out to process messages, ideally all those messages of the company and all communications passing through the organization. Moreover, we know that it even leaves its mark on the cognitive and emotional behavior of individuals.

We believe that the previous paragraphs, though abstract, have made apparent the great difficulty, generally much greater than that customarily acknowledged in reference works, involved in doing what must be done: namely, in raising in a coherent manner the level of complexity of the organization, and not only because of the major comparative increase in strictly technological complexity analyzed in section 5.

At the same time, it is a fact that the functional variety and the capacity of office tools grow unchecked. Consequently, it seems essential for the office technology industry and other technological agents to take the steps necessary to reduce the degree of visible complexity of their products and services. The meeting point between these two sets of actions is established by the law of variety.

In short, simultaneous actions must necessarily be taken on two complementary fronts for innovation to actually take place.

For the sake of example, below we provide an idea of the panoply of different general strategies that could be adopted by each company as well as by the technological industry. The reader will readily know which of these strategies will broaden and which will reduce variety.

a) Each company, for its part, must:

- design, transform and explain its organization in keeping with the new tools, converting its classical bureaucratic structure into a "more pliable" and decentralized structure;
- select and train its personnel, giving priority to such characteristics as personal initiative, self-management, a sense of timing and coordination and an interest in learning the workings of the company and the environment in order to take on more fulfilling jobs;
- devote the amount and type of training necessary to teach each office technological application/function, depending on the degree of complexity (the three-level model could serve as a general guide);
- make the conviviality of hardware and software viewed as a whole a primary criterion for making decisions about office technology material purchases;
- organize an in-house human technical assistance network to resolve all the personal difficulties users encounter during the learning process.

The last three aspects form part of what we call "organizational convivialitation."

- b) Some of the many options the office technology industry has in its hand to contribute to this process include:
- strengthening the naturalness of man-machine interfaces;
 - advertising its offer of products and services with less fantasy and more emphasis on how said offer can solve real organizational problems;
 - selecting and not overmultiplying the functional options of these products;
 - developing or promoting the development of sociotechnical methodologies for applying office technology;
 - simplifying technical language when possible.

6.2 Individuals versus Office Technology

The foregoing equally applies on a small scale to each individual of the company or user of technology. The general technological innovation process can be broken down into as many microprocesses as there are users, subject to the personal reactions of each user and the winning or losing situation created for the user by the changes that accompany technology. To a large degree, said microprocesses are cognitive in nature.

From a cybernetic viewpoint, each microprocess is designed to provide the user with a technological office tool that amplifies (symbolized by the triangle) his capacity to relate on an information basis with his organization and his socioeconomic environment (see the second part of Figure 5). The upper part of Figure 5 depicts the step prior to reaching this state, involving the application of mechanisms governed by the law of variety, namely:

- (a) A reduction in variety (as perceived by the user) of the technological office tool = good user interfaces + organizational convivialitation.
- (b) A broadening of the very complexity of the user in relation to the functionality of the tool = learning, training or recycling.

The equation expressed above figure naturally takes into account the fundamental role of the organization in guiding and catalyzing the cognitive processing symbolized in the amplifier (b), an aspect already addressed above among the strategies ascribable to companies.

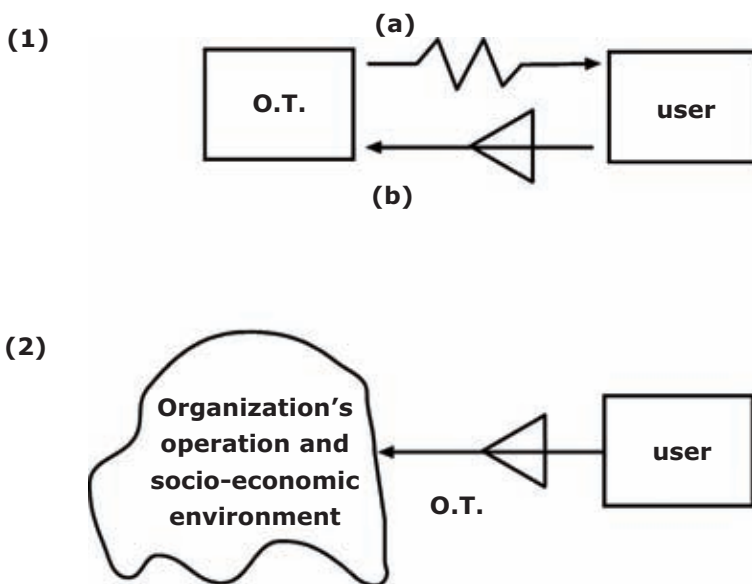


Figure 5

Undoubtedly, the table of measures proposed in this section also contains a considerable number of unknowns and problems to unveil and resolve. Though many of them are beyond the reach of companies, they can be tackled by industry or research institutions, and their solutions will enhance the feasibility of some of the above strategies.

The search for methods of teaching computer word processing to active users, designed to conceal at the outset and gradually reveal as the student progresses the functional and operative complexity of technological office tools, is only a small example of what we want to say (cf. a sample of the work of J.M. Carroll, 1985). In keeping with this same desire to optimize the teaching of beginners, Sáez-Vacas and De la Torre (1991) have begun some work on commercial word processors.

Lastly, there is also room for a variety of apparently modest efforts by other agents outside of a single company, companies as a group and the technological industry, efforts which nevertheless help to improve the innovation processes within the framework of ideas described herein. A book the author has written and recent published (Sáez-Vacas, 1990 (2)) provides an example of this point. The content and structure of this book are designed to act like variety amplifiers and reducers (see Figure 3) in connection with the relationship between managers and office technology.

7. Conclusions

It has been possible to provide a wide-reaching model of office automation by means of a multidimensional theory of three levels of complexity.

It has been shown that one of its dimensions, namely that of the Human and Social Factors vector, is key to the design of office systems, since, when all is said and done, these systems belong to a very special kind of systems called "human activity systems."

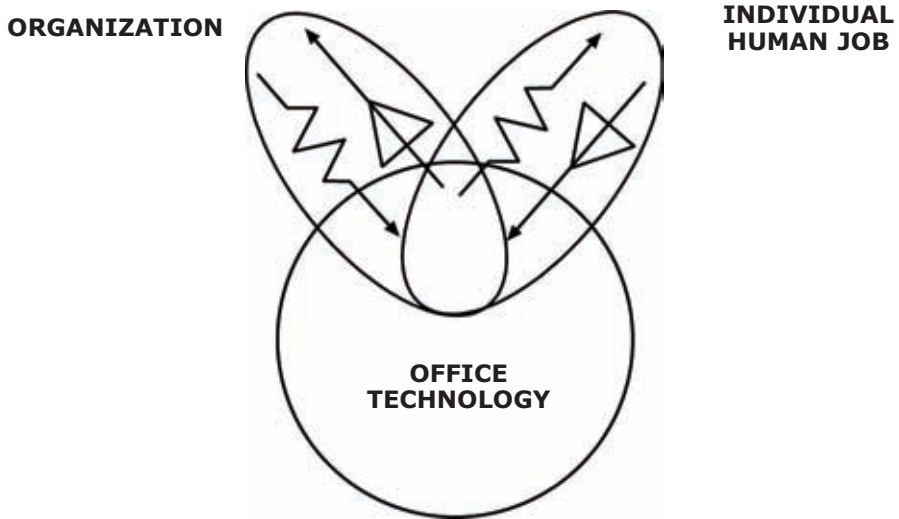


Figure 6

Thus, the three levels of complexity of the Human Factors vector constitute a decisive problem in processes to technologically innovate companies through office automation. Specifically, the third level of complexity seems to be a nucleus that should be isolated if it is to be controlled.

Indeed, once this nucleus has been broken down into three relatively different areas of complexity, a fairly complete conceptual attack, using the cybernetic law of variety, can be launched with a view to its control.

As is summarized in Figure 6, this law is a performance guide on the mechanisms for adjusting complexities, at the collective level of an organization as well as at the individual level where a single person performs his job. This second level, however, is naturally comprised within the strategies of the first level. It is in the

field of the adjustment mechanisms where there is a genuine need for research and ideas.

Controlling the third level complexity of office systems would, in the author's opinion, be tantamount to paving the road of innovation with this new technology.

8. References

Ashby, W.R., **An Introduction to Cybernetics**, Chapman and Hall, London, 1956.

Beer, S., **Designing Freedom**, John Wiley & Sons, London, 1974.

Beer, S., **Diagnosing the System for Organizations**, J. Wiley & Sons, Chichester, 1985.

Carroll, J.M., Minimalist Design for Active Users, **Human-Computer Interaction -Interact '84**, pp. 39-44, North-Holland, Amsterdam, 1985. Reprinted in, Ed. R.M. Baecker, W.A.S. Buxton, **Readings in Human-Computer Interaction. A Multidisciplinary Approach**, Morgan Kaufmann, Publ., CA., 1987.

Checkland, P., **Systems Thinking, Systems Practice**, J. Wiley & Sons, Chichester, 1981.

Flood, R.L., Complexity: a Definition by Construction of a Conceptual Framework, **Systems Research**, vol. 4, 3, pp. 177-185, 1987.

Hirschheim, R.A., **Office Automation: A Social and Organizational Perspective**, J. Wiley & Sons, Chichester, 1985.

Illich, I., **Tools for Conviviality**, Harper Row, New York, 1973.

Mélèse, J. **Approches Systémiques des Organisations**, Ed. Hommes et Techniques, Suresnes, 1979.

Sáez-Vacas, F., Facing Informatics via Three Level Complexity Views, **X International Congress on Cybernetics**, Namur, pp. 30-40, 1983.

Sáez-Vacas, F., Alonso, G., Proposal of a Three Level Complexity Model for Office Automation, **33rd Annual Meeting of the International Society for General Systems Research**, Edinburgh, 1989.

Sáez-Vacas, F., **Complex Office Automation**, Ed. Fundesco, Madrid, 1990 (in Spanish).

Sáez-Vacas, F., **Microinformatics for Managers**, CDN, Madrid, 1990 (in Spanish).

Sáez-Vacas, F., De la Torre, C., A Comparison of the two most popular Text Processors from the Point of View of their Conviviality, paper submitted to the Conference "**Human Jobs and Computer Interfaces**", Tampere (Finlandia), 1991.

Strassmann, P.A., **Information Payoff. The Transformation of Work in the Electronic Age**, Free Press, N.Y., 1985.

Habitualmente ignorada o minusvalorada, la complejidad es un factor multidimensional y ubicuo en el mundo sumamente artificial en el que vivimos. Citemos a Pagels: “la gran frontera inexplorada es la complejidad”; a Whitehead: “toda realidad es unidad compleja”; a Gravitz: “el estudio de la complejidad organizada es el problema clave de todas las ramas de la ciencia moderna”; y al Club de Roma: “el aprendizaje innovador es el instrumento para salvar la distancia entre la creciente complejidad del mundo y nuestra capacidad para hacerle frente”.

Según el autor de este libro necesitamos lograr un conocimiento suficiente de la complejidad, entendida como marco conceptual a ser posible socialmente contextualizado, para abordar la realidad de manera integral. En particular, ese conocimiento resulta imprescindible para los técnicos en infotecnologías, donde la complejidad emerge como una dimensión central, al tiempo que sus impactos se extienden a todos los ámbitos sociales.

Fernando Sáez Vacas, profesor en esta Escuela desde 1967, catedrático por oposición de Ordenadores Electrónicos desde mayo de 1974 y profesor emérito desde octubre de 2009, ha desarrollado una dilatada investigación personal en el campo de la sociotecnología de la información y postula la necesidad de crear unas bases de sociotecnocultura basadas indispensablemente en la complejidad, concebida en sentido amplio.