

658.4078

G 633

UNIVERSIDAD DE ATACAMA
FACULTAD DE INGENIERIA



P.T.D. INGENIERIA CIVIL INDUSTRIAL UDA-ANDINA

ASIGNATURA

"SISTEMA DE INFORMACION ADMINISTRATIVOS"



PROFESOR : RAUL GOMEZ GONZALEZ

MARZO DE 1997

CONTENIDOS

- A. El enfoque de sistemas y la teoría general de sistemas.
 - 1. Introducción : la necesidad de una Base teórica.
 - 2. Hacia el enfoque de sistemas.
 - 3. Hacia una teoría General de Sistemas.
 - 4. El enfoque de Sistemas.
 - 5. Teoría General de Sistemas.

- B. Los Sistemas Organizacionales.
 - 1. Algunos aspectos de la teoría de la regularización.
 - 2. La Homeostasis y los factores de control.
 - 3. Un modelo de Coherencias Operacionales Homeostáticas.
 - 4. La ley del requisito de variedad de ashby.
 - 5. La propuesta de Stafford Beer.

- C. Los Sistemas Informáticos en la Organización.
 - 1. El rol de los Sistemas Informáticos en la Organización.
 - 2. Desarrollo de Sistemas Informáticos.
 - 3. Análisis y Diseño Estructurados.
 - 4. Evaluación y Administraración del proceso de Desarrollo de Software.

SECCIÓN A

***EL ENFOQUE DE SISTEMAS Y LA TEORÍA GENERAL DE
SISTEMAS***

1. INTRODUCCIÓN: LA NECESIDAD DE UNA BASE TEÓRICA.

El campo de aplicaciones de los computadores digitales en el contexto organizacional ha pasado a ser conocido modernamente bajo diversas denominaciones: Sistemas de Información, Informática, Sistemas de Información Integrados, etc. Si se busca en la literatura definiciones o conceptos precisos de qué es lo que hay detrás de tales nombres, se verá que predomina la vaguedad y la ambigüedad, y que existen grandes diferencias de opiniones. Si se profundiza más en la terminología, se verá que conceptos tales como sistemas, información, administración, y organización se utilizan, en muchos casos, en nombres y definiciones como conceptos primitivos, es decir, como si no permitieran una adecuada caracterización. Esto no sólo tiene importancia desde el punto de vista semántico. Las más graves consecuencias de la falta de conceptualización tienen que ver con el proceso de diseño o desarrollo de sistemas. En efecto, si no se puede caracterizar algo, malamente se lo puede diseñar y construir en forma eficiente.

En la práctica, la poca comprensión que existe acerca de los Sistemas de Información y su papel dentro de la Organización, ha llevado a múltiples problemas; entre otros: el uso de los computadores en aplicaciones que no son las más rentables, una actitud desfavorable por parte de los usuarios hacia los Sistemas de Información, y el no aprovechamiento integral de tales sistemas como instrumento para hacer más racional la administración de empresas e instituciones.

De aquí la necesidad de establecer ciertas bases que permitan una sólida conceptualización de los Sistemas de Información. Esto permitirá reponder en forma precisa preguntas tales como:

- ¿Qué son los Sistemas de Información?
- ¿Cuáles son sus componentes?
- ¿Qué papel juegan en una empresa o institución?
- ¿Qué factores influyen sobre su desarrollo?
- ¿De qué depende la efectividad de ellos en el contexto de una Organización?

Sin embargo, éste no es el único tipo de interrogantes que se deben responder en relación a los Sistemas de Información. Suponiendo que se tiene claro qué es lo que se desea diseñar y desarrollar, queda el problema de definir cuál es la mejor manera de llevar a cabo este proceso; vale decir, cuál es la metodología que se debiera usar.

Al respecto, la experiencia es mala, ya que, por falta del uso de metodologías adecuadas, el proceso de desarrollo de sistemas ha conducido a muchos problemas, tales como mala interpretación de las necesidades de los usuarios de los sistemas, emisión de gruesos informes computacionales que nadie usa, mala documentación, etc.

Se aceptará como premisa que un Sistema de Información es, en general, algo complejo que involucra personas y máquinas y tiene muchos componentes y numerosas relaciones entre ellos. Se debe hacer notar que tal tipo de sistema existe para cumplir un cierto objetivo o función básica, a diferencia de sistemas naturales que sólo existen; por ejemplo, el sistema planetario. Es conveniente establecer la dicotomía sistemas naturales y sistemas artificiales, siendo los primeros de origen físico y biológico, y los segundos aquellos que el hombre ha

construido para satisfacer sus necesidades. Obviamente la característica más importante de los sistemas artificiales es la participación humana.

Los sistemas artificiales introducen los problemas de diseño y desarrollo. Por el contrario, los sistemas naturales no se diseñan, existen, y sólo es posible tratar de encontrar las leyes que rigen su comportamiento.

En la ciencia clásica (física-biológica) ha predominado el estudio de los sistemas (fenómenos) naturales, y sus métodos, fundamentalmente de análisis, disección, clasificación, muy exitosos en tales sistemas, han influenciado fuertemente los procedimientos y metodologías de disciplinas alejadas de los sistemas naturales, entre otras la Ingeniería y aquella asociada a los Sistemas de Información.

La pregunta que se puede hacer ante esta situación es: ¿Será conveniente, para diseñar sistemas (artificiales) complejos una metodología basada en análisis?. En otras palabras, ¿debe ser el enfoque de diseño uno de división del problema en pequeñas partes, abordando cada parte por separado?.

Varios autores, entre ellos Simón, von Bertalanffy, Ackoff, y otros, han mostrado la inconveniencia de un enfoque de este tipo y la necesidad de una ciencia de lo artificial (o de sistemas), que más que el análisis enfatice la síntesis, lo sistémico, la organización, las relaciones entre las partes, etc.

Estos autores no se han quedado en una crítica al enfoque analítico, sino que han propuesto enfoques alternativos, de los cuales se derivan o pueden derivarse metodologías de diseño de sistemas complejos artificiales, como los Sistemas de Información.

Resumiendo, de lo dicho se desprende que se debe buscar y estudiar conceptos que ayuden a caracterizar y definir los Sistemas de Información y su rol en las Organizaciones, y nuevos enfoques teóricos de diseño de donde se pueda derivar ayuda metodológica.

El lugar lógico donde se debe empezar este estudio es, entonces, aquellas teorías y conceptos que permitan caracterizar y diseñar sistemas artificiales en general. Luego debe especializarse estas teorías o conceptos a los Sistemas de Información. Sin embargo, por ser los Sistemas de Información subsistemas de sistemas sociales u Organizaciones, esta especialización se hará pasando primero por estos sistemas más generales.

2. HACIA EL ENFOQUE DE SISTEMAS.

Hablar de sistemas no es referirse a una noción que se ha desarrollado sólo en los últimos años. Es posible afirmar que la idea de sistemas se perfila con rasgos distinguibles ya en el saber del siglo V a. c., y su cuerpo de conceptos se fue perfilando desde los comienzos de la filosofía y de la ciencia, que en cuanto a expresiones del conocimiento se caracterizan fundamentalmente por su lucha constante contra toda forma de ignorancia y superstición, determinantes estas últimas en gran medida de la cultura de los primeros hombres, quienes, al observar el mundo en que vivían, lo percibieron como un entorno hostil en el cual no era posible un orden lógico, asumiendo como consecuencia de ello que todos los fenómenos ocurrían mas bien por la influencia directa de fuerzas que estaban más allá de su propia realidad, razón por la cual, para influir sobre aquello que no les favorecía, recurrieron simplemente a la supuesta eficacia de la magia.

Frente a estas limitaciones, se puede decir que la filosofía, y también la ciencia, nacieron cuando el hombre comenzó a percibir la realidad como una unidad ordenada, y por ende inteligible, en donde justamente por ello se hace susceptible de control y modificación a través del pensamiento y la acción racional que es posible derivar.

Un ejemplo de este desarrollo lo constituyen las explicaciones en torno al arco iris. Este es uno de los tantos fenómenos cautivantes de la naturaleza. Es de hermoso colorido, tiene forma regular, y constituye un espectáculo majestuoso. Durante mucho tiempo se explicó dicho fenómeno por la acción de una fuerza sobrenatural: la divinidad. A pesar de las diferentes interpretaciones religiosas o sobrenaturales acerca del arco iris, muchos estudiosos de los fenómenos del universo, desde tiempo inmemorial, se esforzaron por hallar una explicación natural a dicho fenómeno. Se observó que la coloración irisada podía contemplarse en las gotas de rocío, en los prismas hexaédricos, en la nube de agua de las cascadas, en las salpicaduras que levantaban los remos, etc. Al comparar estos casos se vio que la única circunstancia en que todos concuerdan es que la luz pasa a través de un medio transparente de forma esférica o prismática. Como quiera que el arco iris sólo aparece cuando llueve y hay sol, se supuso que era producido por el paso de la luz a través de un medio transparente de forma esférica, es decir, a través de gotas de agua. La conclusión resultó verdadera.

Sin embargo, el conocimiento, configurado en sus comienzos en mayor o menor grado de este modo, alcanzó en la historia de su desarrollo un carácter científico sólo en la medida en que fue descubriendo leyes y adquiriendo fuerza previsoras.

De esta forma, se puede decir que la tarea del conocimiento científico consiste en descubrir la naturaleza de las cosas y los principios conforme a los cuales obra. Más aún, se puede agregar que trata la cosas y los sucesos del mundo tal como los encuentra, y que arranca de la necesidad y el deseo que el hombre tiene de prever y controlar su medio ambiente.

Es así como surge un sistema de conceptos acerca de los fenómenos y leyes del mundo externo, en cuya perspectiva, al analizar las ideas vertidas principalmente por los pensadores de mayor relevancia, a partir de los griegos del siglo VI a.c., es posible encontrar, en no pocas de ellas, y expresados con mucha claridad, aspectos esenciales asociados a la noción de sistema.

Al respecto, Ludwig von Bertalanffy, en su trabajo "Historia y Situación de la Teoría General de Sistemas", destaca, entre otros aspectos, que las propiedades y naturaleza de los fenómenos en los niveles superiores, no son deducibles de las propiedades y naturaleza de sus procesos componentes, si éstos se toman aisladamente, lo cual hace necesario considerar explícitamente las relaciones que los ligan. Todos estos planteamientos los sintetiza Bertalanffy con la frase aristotélica *el todo es más que la suma de sus partes*, perspectiva que corresponde a una visión de la realidad que incluso hasta hoy día sigue siendo la expresión del problema sistémico fundamental.

Aunque estas ideas tuvieron significativa relevancia en la época aristotélica, fueron obviadas en los desarrollos posteriores de la ciencia occidental, de tal forma que no hubo una preocupación explícita por dilucidar problemas fundamentales que se manifiestan en muchos fenómenos de la realidad objetiva, tales como organización, integridad, adaptación, y desarrollo.

Sin embargo, surgieron algunos pensadores que en sus trabajos utilizaron explícitamente las nociones de muchos de los conceptos básicos sobre los que se sustenta el enfoque de sistemas.

El místico Dionisio de Arepagita, por ejemplo, establece un orden jerárquico en el marco resultante de la relación entre lo que él identifica como "coros de ángeles" y los organismos de la Iglesia.

La noción de *coincidencia oppositorium*, entendida como la lucha de las partes dentro de una totalidad y de las que invariablemente surge una unidad de orden superior, introducida por Nicolás de Cusa, es otro ejemplo del empleo de nociones básicas de sistemas.

Leibnitz, en su trabajo "Mathesis Universalis", habla de una matemática que no consiste solamente de expresiones numérico-cuantitativas, sino que también debe permitir la representación formal de conceptos derivados del acto de pensar. Bertalanffy destaca también como aporte relevante el trabajo de Hegel y Marx, en cuanto al carácter dialéctico de la estructura del pensamiento y del mundo que lo engendra; es exclusivamente profunda en ellos la afirmación de que el reflejo adecuado de la realidad lo ofrece no un juicio aislado, sino sólo la unidad de los dos lados de la contradicción lograda en el proceso dialéctico: tesis - antítesis - síntesis.

El planteamiento de organizaciones de orden superior de Gustavo Fechner sugiere la existencia de lo que él denomina *organizaciones supra-individuales*, entendidas como instancias de orden superior al de los objetos que el hombre, en su experiencia cotidiana, aprehende como unidades. En este contexto se habla de comunidades de vida, donde la tierra es una unidad determinada por la interacción de sus componentes.

Es posible constatar así que la forma de concebir la realidad a la que en los últimos años se ha denominado *enfoque de sistemas*, no es de reciente aparición, y más aún, se han mantenido en el tiempo, como el propio Bertalanffy lo señala cuando expresa que: "los problemas con los que los científicos de nuestros días tropiezan debido al concepto de *sistema* surgieron no de repente, y no son el resultado exclusivo del desarrollo contemporáneo de la Matemática, las ciencias naturales, y la técnica, sino que sólo son la expresión actual de problemas que durante siglos han estado planteados a los científicos, y

han sido tratados previamente con lenguajes diferentes". No obstante, es preciso puntualizar que estos planteamientos no fueron privilegiados en el desarrollo de la ciencia.

Más allá de los aspectos económicos asociados a la Revolución Industrial de los siglos XVI y XVII, se tiene que como consecuencia de ella se dejó de privilegiar la concepción aristotélica de la realidad holista de carácter descriptivo-metafísica, por la matemático-positivista o galileana, surgida a mediados del siglo XVI, en donde aquella realidad se percibe como una cadena de relaciones causales, es decir, de causa-efecto, regida por leyes matemáticas.

Ello no significó en absoluto que el fundamento del planteamiento de Aristóteles, sintetizado, tal como se indicó, de que el todo es mayor que la suma de las partes, haya perdido totalmente su relevancia, más aún cuando la organización de un todo real, derivada de la relación entre sus partes, trascendiéndolas, al considerarlas individualmente, es una propiedad objetiva que el hombre descubre, comprende, y refleja en su conciencia durante el proceso de adquisición del conocimiento. Obviamente ello no significa que sea una construcción arbitraria de nuestra conciencia. Al respecto, Bertalanffy dice: "Es un hecho con el que nos enfrentamos cada vez que observamos un organismo vivo, un grupo social, e incluso un átomo".

Pero aún así, la ciencia continuó con la idea de fragmentar todo problema en tantos elementos aislados como sea posible. Estas ideas son aquellas que se privilegian preferentemente en los métodos que utiliza la ciencia en los siglos XVII y XVIII. Un ejemplo de lo ocurrido queda de manifiesto al analizar las premisas en las que se apoya Descartes, quien en su obra "Discurso del Método", propone *la división de cada una de las dificultades con que tropieza la inteligencia al investigar la verdad, en tantas partes como fuera necesario para resolverla*.

Si se observa lo planteado por Descartes, se destaca la idea de dividir todo fenómeno en tantas partes simples e independientes como sea posible, con el fin de contar con una base para el establecimiento de procedimientos eficaces para su estudio y comprensión.

En general se puede afirmar que los denominados métodos elementalistas y mecanicistas fueron los de mayor arraigo en las ciencias hasta comienzos del siglo XX, predominando un estilo de pensamiento basado en el criterio de que toda teoría científica debía caracterizarse por un sistema de leyes de rígida determinación, y que el método analítico permitiría descubrir los *elementos* que llevarían a un conocimiento verdadero del todo.

El análisis de algunas de las características fundamentales de esos métodos y ese estilo de pensamiento, que representan una etapa histórica del desarrollo científico al cumplir una función cognoscitiva determinada, permite comprender mucho mejor las causas que motivaron al surgimiento del enfoque sistémico.

El *método elementalista* cuenta con un largo y fecundo recorrido. Su surgimiento se produce en las etapas iniciales del desarrollo histórico, y en él queda plasmada la convicción del que el conocimiento de cualquier fenómeno implica la búsqueda y el establecimiento de los fenómenos primarios, de los datos básicos que constituyen y determinan dicho fenómeno, pues solo en el análisis de éstos se puede encontrar las características fundamentales del objeto en su totalidad.

Para este método el problema de la integridad de lo complejo se resuelve con la reducción de lo complejo a lo simple, del todo a la parte. Esta es la razón por la cual este método ha recibido la denominación de *método reduccionista*. Las dificultades derivadas del hecho de que no siempre fuese posible establecer estos "ladrillos" iniciales de los diferentes fenómenos fue interpretada como el resultado de las debilidades propias del conocimiento humano, o bien como deficiencias provenientes de los métodos empleados para esa búsqueda.

La realidad existencial de estos elementos no se ponía en duda; y esta convicción, a la par que estimuló una profunda investigación de la composición elemental o *elementalista* de los fenómenos, creó una visión unilateral de éstos y, en general, del universo, proyectando el potencial de investigación de la ciencia por canales muy estrechos.

Esto tuvo lugar principalmente en la física, la biología, y la psicología, ciencias en las que la búsqueda de ese elemento básico fue durante mucho tiempo una de las tareas centrales de los investigadores.

Es necesario, no obstante, hacer justicia al método citado, no sólo reconociendo que gran parte de los descubrimientos de la ciencia se han llevado a cabo con su concurso, sino comprendiendo también las causas objetivas que los determinaron, entre las que se pueden destacar las siguientes:

- Una de las formas más asequibles de abordar el estudio de un objeto complejo consiste, precisamente, en la descomposición de sus partes integrantes, en el estudio por separado de estas partes, con el fin de emprender luego su síntesis y descubrir las leyes que vinculan estas partes en un todo.

La descomposición de todos los objetos sometidos al estudio de la ciencia permite que en un momento determinado se pueden encontrar elementos comunes en fenómenos diferentes, no sólo en los marcos de una misma área de la ciencia, sino en objetos de áreas diferentes. Por ejemplo, el descubrimiento de las leyes dinámicas de la mecánica clásica y su manifestación en diferentes esferas de la realidad, llevó a la elaboración de un modelo de ley científica que se consideró vigente para toda la esfera del mundo objetivo y que, en su aplicación inicial, sin duda dio un impulso positivo a numerosos avances científicos.

- Es un hecho de que en cierta medida, y a pesar de los cambios ocurridos en la ciencia, este método no ha agotado aún sus posibilidades y sigue cumpliendo una determinada función cognoscitiva, en cooperación con los nuevos métodos.

Un ejemplo muy significativo de esto ha sido el surgimiento de la Cibernética, determinado en parte por una ampliación del aparato conceptual del elementalismo clásico; el descubrimiento de que la *información* es un elemento común a todos los fenómenos, de rango similar a los conceptos de *sustancia* y *energía*, antes fundamentales para la *ciencia clásica*.

El método *mecanicista*, por su parte, se identifica en muchos aspectos con el anterior, y en esencia está basado en un mismo estilo de pensamiento científico, pero centra su atención en la búsqueda de un principio único de vinculación de los elementos, y reconoce sólo un tipo de relación de causa y efecto: aquellas que se caracterizan por la univalencia de los vínculos y la equivalencia de los parámetros.

En el siglo pasado, el método mecanicista adquirió gran influencia al vincularse estrechamente con las ciencias físico-matemáticas. Fue precisamente la mecánica clásica la primera teoría científico-natural que pudo formular sus leyes fundamentales en el lenguaje riguroso de las matemáticas; durante cierto período ésa se consideró como el modelo de la ciencia, por la precisión de sus conclusiones teóricas, de las que, de hecho, quedaron eliminados todos aquellos tipos de vínculos que no respondían a los criterios de rigurosidad.

Naturalmente, estos dos métodos no fueron los únicos. Conocida es la constante polémica y, a veces, la enconada lucha que tuvo lugar en el seno de la ciencia entre estos dos métodos y el denominado *integralista*, que negaba la posibilidad de reduccionismo, e indicaba, no sin razón, que el *todo* posee cualidades que no se presentan en ninguna de sus partes. Este método desempeñó un papel positivo en la historia de la ciencia, vinculado, eso sí, frecuentemente a concepciones filosóficas idealistas, razón por la cual no pocas veces se orientó hacia la búsqueda de las particularidades específicas del todo en la esfera de lo sobrenatural.

Con el progresivo avance de la ciencia, los problemas que eran objeto de estudio fueron siendo cada vez más complejos, es decir, los problemas que había que enfrentar estaban constituidos por muchos elementos organizados y en interacción dinámica. Esto queda de manifiesto cuando, a pesar del enorme éxito obtenido en la explicación de múltiples fenómenos del mundo físico, no se logró explicar el funcionamiento de los sistemas como tales. Así, no lograron explicar ciertos problemas mecánicos en que intervienen varios cuerpos, ni los sistemas biológicos, ni los sistemas sociológicos.

Consciente de este problema, la ciencia enfrentó su solución mediante dos ideas centrales. Una de ellas fue la comparación con máquinas hechas por el hombre, y la otra, la idea del orden como producto del azar.

- El primer planteamiento fue sugerido por Descartes en el siglo XVII en su obra filosófica "Tratado del Hombre", en la que considera a este último como una *máquina viviente*. Todas las funciones propias de semejante máquina: la digestión, el latido del corazón y las arterias, la alimentación, el crecimiento, y la respiración, son consecuencias naturales de la distribución de los órganos de la máquina mencionada, lo mismo que los movimientos del reloj o de un autómatas son el resultado de la acción de contrapesos o ruedas.

Descartes escribía: "Trataré de explicar el mecanismo de nuestro cuerpo de forma que tengamos tan poco de referir al alma los movimientos no relacionados con la voluntad, como no lo tenemos para considerar que los relojes tienen un alma que les obliga a dar la hora".

En esta misma línea de pensamiento se puede situar el trabajo de J. La Mettrie, médico y pensador francés del siglo XVIII, quien editó en 1748 un libro titulado "El Hombre Máquina", que fue quemado públicamente por su tendencia materialista y atea. En él se fundamenta la idea de que el alma no puede ser algo distinto al cuerpo, sólo existe la materia, que se mueve según las leyes de la mecánica.

- El segundo planteamiento corresponde básicamente a la idea *darwiniana* de la selección natural.

Estos enfoques no dejaron de tener éxito. De este modo se fueron generando las primeras explicaciones de los fenómenos biológicos, fundamentalmente en lo relativo al funcionamiento fisiológico de los órganos, y a las estructuras y procesos enzimáticos de la célula, derivados por los estudios generados en el siglo XVII, y de las explicaciones logradas al concebir al organismo como una máquina calórica, quinodinámica, celular, y cibernética.

Tan maravillosos autómatas permiten constatar una tendencia a copiar o a modelar de un cierto modo el comportamiento de los seres vivos. Cabe señalar que semejante tendencia también es propia de la actualidad.

Del mismo modo, el entendimiento de la conformación *organizada* de un organismo a partir de ciertos fenómenos aleatorios, se constituye en un método también de cierto éxito en la explicación de la genética molecular y de la biología en general, conocido con el nombre de Teoría Sintética de la Evolución.

Sin embargo, y a pesar de estos avances, quedaron muchos problemas fundamentales sin respuesta.

Bertalanffy señala que "La máquina animal de Descartes era un modelo que explicaba el admirable orden de los procesos observados en los organismos vivos. Pero, según Descartes, la máquina tenía a un Dios como creador. La evolución de las máquinas a través de procesos azarosos parece ser más bien autocontradictoria. Los relojes de pulsera o las medias de nylon no se encuentran por regla general en la naturaleza como resultado de procesos aleatorios, y ciertamente las *máquinas* mitocondriales de organización enzimática, incluso en la más simple célula o molécula nucleoproteica, son incomparablemente más complejas que un reloj o los sencillos polímeros que forman las fibras sintéticas. *La supervivencia del más apto* (o la *reproducción diferencial* según la terminología moderna) parece conducir a un argumento circular. Éste requiere la existencia de organismos que se automantengan, existencia previa a la participación de éstos en una competición en que predominarán aquellos con un valor selectivo o reproducción diferencial más alto. Este automantenimiento, sin embargo, es un postulado: no lo explican las leyes ordinarias de la Física. Al contrario, la Segunda Ley de la Termodinámica señala que los sistemas ordenados en los que ocurren procesos irreversibles tienden hacia los estados probables, y por tanto hacia la destrucción del orden existente, y, en último término, a la decadencia".

En consecuencia, se puede decir que el enfoque analítico-mecánico, si bien significativo en importancia para la ciencia, fue más bien limitado en ciertos aspectos fundamentales.

3. HACIA UNA TEORÍA GENERAL DE SISTEMAS.

Se puede decir que en la década del 30, en virtud de los trabajos de Bertalanffy, se empiezan a apreciar con claridad las propuestas de los que se ha venido a denominar **Enfoque de Sistemas**, en contraposición a los métodos clásicos de la ciencia. Ello queda de manifiesto principalmente en Biología, Psicología, y en las ciencias sociales, destacándose como un factor de relevancia la idea de que ciertas propiedades de los sistemas no dependen de su naturaleza específica, sino que son comunes a los más diferentes sistemas, sentando las bases de lo podría considerarse como una teoría general, la **Teoría General de Sistemas**.

Estas propiedades, que se repiten en diversos sistemas, se entendieron en un comienzo tan sólo como semejanzas, la mayoría de las veces geométricas, pero también cinemáticas o termodinámicas.

Se afirmaba que dos sistemas son *semejantes* cuando las variables que determinan a uno eran de la misma naturaleza física que las que determinaban al otro, y cuando en la misma correlación temporal, los valores que asumen dichas variables son proporcionales.

Posteriormente se sobrepasó la noción de semejanza, al considerar en los sistemas a variables de diferente naturaleza física, relación establecida como *analogía* entre sistemas, la cual se expresa normalmente en términos de ecuaciones algebraicas o diferenciales.

Así entonces, se consideran análogos dos sistemas si las ecuaciones que definen a ambos son similares. En este contexto, se concibió lo que se conoce como Teoría de la Semejanza. Ello ocurrió cuando se constató que era posible que cierta disciplina que estudia determinados sistemas podía utilizar métodos desarrollados por otra disciplina que enfrenta sistemas de diferente naturaleza.

Un ejemplo característico de esta tendencia lo constituye la utilización de los procedimientos de análisis de los circuitos eléctricos a otros sistemas, tales como circuitos mecánicos o térmicos. Incluso se configuró lo que se denomina Teoría de los Circuitos Generalizados.

Si se analiza en profundidad estas tendencias, queda claramente perfilado un cambio sustantivo, por ejemplo, cuando se pasa de la semejanza geométrica a otros tipos de semejanzas, desarrollo que se puede considerar como un primer paso importante para la conformación del concepto de *Sistema General*, así como del mismo modo, la proyección de la semejanza a la noción de analogía entre sistemas, constituyó un segundo avance.

La relación de semejanza es reflexiva, simétrica, y transitiva, por lo tanto es en sí una relación de equivalencia, por lo cual, una vez establecida, es posible dividir todos los sistemas ligados a un cuerpo disciplinario específico, en clases de equivalencia. Entonces, mediante lo que se puede llamar un sistema típico, es posible representar cada una de las clases de equivalencia. Así, los resultados de las investigaciones en torno al sistema tipo, mediante el empleo de las reglas de la Teoría de la Semejanza, se pueden aplicar a cualquier sistema perteneciente a la clase de equivalencia.

Si se observa con atención, se puede apreciar que en el concepto de analogía subyace una relación de equivalencia, pues la analogía es una generalización de la equivalencia, pero se expresa en términos tales que los elementos de cierta clase de equivalencia son sistemas cuyo origen está en disciplinas diferentes.

Por lo tanto, la noción de analogía también introduce una relación de equivalencia; la diferencia radica en que los elementos de la misma clase de equivalencia son ahora sistemas provenientes de distintas disciplinas científicas.

De esta forma entonces, los resultados de los estudios en torno al sistema tipo, representante de una clase de equivalencia, pueden transferirse a otras disciplinas.

Del mismo modo, se puede decir que cuando ciertos isomorfismos¹ son susceptibles de generalizar para cualquier relación, constituyéndose en homomorfismos², se está frente a todo el sentido que tiene el concepto de Sistema General.

Así, el Sistema General es un modelo formal que se genera cuando una relación de equivalencia del tipo isomorfa, se aplica a ciertas características de los sistemas.

Dos sistemas se consideran isomorfos cuando los elementos y los vínculos y relaciones entre ellos se encuentran en un sistema en correlación unívoca mutua con los elementos, vínculos, y relaciones de otro. Es por esta razón que los sistemas polimorfos³ pueden describirse por medio de estructuras matemáticas idénticas. Esta circunstancia ha adquirido ahora gran importancia en la modelación lógico-matemática, fundamentalmente para aquella realizada por medio de computadores.

Cuando se modela en un computador, se realiza el polimorfismo entre la estructura matemática del objeto investigado y el programa de la máquina, expresado en un algoritmo numérico mediante un lenguaje de programación. En principio, cualquier dependencia matemática puede representarse en forma de un determinado algoritmo numérico, y el proceso de cálculo con la programación correspondiente se hace isomorfo a cualquier sistema dinámico.

En base al isomorfismo de la estructura de las relaciones y por medio de la modelación lógico matemática, se pueden conocer diferentes niveles de organización de la materia. I. A. Urmantsev demostró que *cualquier objeto de la naturaleza es un representante de cierto conjunto polimórfico*.

Esta conclusión abre grandes posibilidades a la búsqueda científica de modelos, a partir de la convicción objetiva sobre la existencia de un determinado isomorfismo entre los diferentes sistemas de la naturaleza.

En la modelación es muy frecuente, sobre todo después de la aparición del método funcional de la cibernética, la utilización de una exigencia menos rigurosa de comunidad entre el modelo y el original: la relación de homomorfismo.

En el caso del homomorfismo, las relaciones entre dos sistemas no son mutuamente unívocas. El matemático G. Polya dio una definición de homomorfismo que resulta interesante destacar:

"El homomorfismo es una especie de traducción sistemática abreviada. El original no sólo se traduce a otro lenguaje, sino que se abrevia, de tal forma que, en última instancia, lo que se obtiene después de la traducción y lo abreviado, resulta sistemáticamente igual a una mitad o

¹Isomorfismo: Correspondencia biunívoca de las estructuras matemáticas de los sistemas.

²Homomorfismo: Isomorfismo que se cumple en un nivel de detalle más profundo.

³Polimorfismo: Isomorfismo que se cumple en todo nivel de detalle.

una tercera parte comprimida o a cualquier otra parte de la extensión inicial. En esta reducción pueden perderse las sutilezas, pero todo lo que está en el original aparece de alguna forma en la traducción, mientras que las correlaciones se conservan, aunque a escala reducida".

Estas perspectivas aparecen como expresiones de algunas declaraciones programáticas que de una u otra forma se han constituido en hitos dentro del desarrollo de lo que se ha denominado **Teoría General de Sistemas**.

Respecto a esta última, Bertalanffy señala:

1. Existe una tendencia general a la integración de las ciencias naturales y sociales.
2. Esta integración parece centrada en la Teoría General de Sistemas.
3. Tal teoría puede constituir un instrumento importante en la búsqueda de la exactitud de las ciencias no físicas.
4. Al desarrollar principios unificadores que atraviesen verticalmente el universo de las ciencias individuales, esta teoría nos aproximará a la meta de la unidad de la ciencia.
5. Este logro conducirá a una integración de la educación científica que es muy necesaria.

En 1954 se fundó la Sociedad para la Investigación de Sistemas Generales, cuyos miembros establecieron un programa de cuatro puntos que habrían de orientar la investigación sobre la Teoría General de Sistemas:

1. Investigar el isomorfismo de conceptos, leyes, y modelos en varios campos, y promover transferencias útiles de un campo a otro.
2. Favorecer el desarrollo de modelos teóricos adecuados en aquellos campos donde falten.
3. Reducir en lo posible la duplicación del esfuerzo teórico en campos distintos.
4. Promover la unidad de la ciencia mejorando la comunicación entre especialistas.

Este proyecto tenía como incentivo tanto lo que Bertalanffy caracteriza como la "situación caótica en la cual se encontraba la investigación teórica en esos años", como también la irrupción de homologías conceptuales de una solidez bastante significativa como para presentar una perspectiva unitaria en la ciencia, caracterizada por un desarrollo deliberadamente fragmentado por las metodologías en uso, que se tenían principalmente en el campo físico.

Un ejemplo de la aplicación de isomorfismos horizontales y verticales en distintos campos de la ciencia son los estudios de Volterra en torno a poblaciones, los cuales fueron utilizados posteriormente en Econometría y en Cinética.

4. EL ENFOQUE DE SISTEMAS.

La noción de sistemas es el fundamento de lo que se ha venido a denominar **Enfoque de Sistemas**. Éste, como señala Joël de Rosnay, "se apoya en una consideración global de los problemas o de los sistemas en estudio y se conceptúa en el juego de interacciones entre sus elementos, en contraposición a los enfoques elementalistas y mecanicistas que, como se señaló en el Capítulo 1, fueron los de mayor arraigo en las ciencias del siglo XIX y comienzos del XX. En ellos predomina un estilo de pensamiento basado en el método analítico el primero, es decir, en la resolución de la integridad mediante la reducción de lo complejo a lo simple, y en el reconocimiento sólo de un tipo de relaciones de causa y efecto en el segundo, de aquellas que se caracterizan por la univalencia de los vínculos y la equivalencia de los parámetros.

En otras palabras, aquellos métodos se basan en el criterio general que sugiere que toda teoría científica debía caracterizarse por un sistema de leyes de rígida determinación, y en lo específico se concebía el objeto de investigación científica, como lo señala George Klir, "como una colección de componentes aislados, de cuyas propiedades intentaban deducirse las propiedades de todo el conjunto, sin considerar las interacciones entre las partes".

A pesar de su inmenso valor teórico y práctico, la relación causal de los fenómenos obviamente no refleja toda la diversidad de las relaciones de los fenómenos distinguidos como unidades compuestas o sistemas.

Muchos pensadores lo pusieron así de manifiesto. El materialismo histórico, por ejemplo, señala que la causalidad no es sino una pequeña partícula de la concatenación universal, que el concepto humano de la causa y el efecto siempre simplifica algo de la conexión objetiva de los fenómenos de la naturaleza, reflejándola tan sólo aproximadamente, aislando artificialmente tales o cuales aspectos del proceso universal único.

Es en la perspectiva de las limitaciones de los métodos analíticos en que el enfoque sistémico se presenta como una alternativa para enfrentar el estudio de los fenómenos que al ser distinguidos aparecen en esencia complejos. Obviamente estos fenómenos pueden ser estudiados por otros métodos, pero, como lo señala Bertalanffy, los resultados obtenidos por aquellos otros métodos no permitirían tener una visión acertada de las unidades distinguidas como sistemas, más allá, incluso, de la utilización explícita o no de la noción de sistema.

El método analítico, que como se dijo, aísla los elementos y las variables, a fin de examinarlos uno por uno, obviando la relacionalidad que se manifiesta entre ellos, no permite explicar cabalmente los fenómenos distinguidos como unidades compuestas.

Para entender, por ejemplo, la actividad del sistema nervioso, no basta con descubrir el trabajo de las células nerviosas de las neuronas y sus nudos, sino que es preciso analizar las relaciones que surgen entre neuronas aisladas en cada proceso nervioso concreto.

Interesante resulta lo que al respecto señala Joel de Rosnay en su trabajo "El Macroscopio". Para él, "el enfoque analítico pretende reducir un sistema a sus elementos constitutivos más simples, a fin de estudiarlos en detalle y comprender los tipos de interacción que existen entre ellos. Después, modificando 'una variable a la vez', deducir leyes generales que

permitan predecir las propiedades del sistema en condiciones muy diferentes; para que tal predicción sea posible, es preciso que puedan aplicarse las leyes de actividad de las propiedades elementales. Este es el caso de los sistemas homogéneos, es decir, de aquellos que consisten de elementos semejantes y presentan interacciones débiles. En este caso, las leyes estadísticas se aplican bien y permiten comprender el comportamiento de la multitud, de la complejidad desorganizada". Luego agrega que "Las leyes de actividad de las propiedades elementales no se aplican evidentemente a los sistemas de elevada complejidad, constituidos por una diversidad muy grande de elementos, ligados por fuertes interacciones. Tales sistemas deben ser abordados por métodos nuevos, como los que agrupa el enfoque sistémico. Su objetivo: considerar un sistema en su *totalidad*, su *complejidad*, y su *dinámica propia*".

A modo de ejemplo, la Simulación se aparece como un método que permite observar el comportamiento de un sistema y observar en tiempo real los efectos de los diferentes tipos de interacciones entre sus elementos. De esta forma, estudiar el comportamiento en el tiempo del sistema permitirá, con niveles adecuados de efectividad, definir las reglas de acción tendientes a su modificación o incluso a su redefinición.

El cuadro comparativo que muestra Rosnay permite tener una visión del contenido y alcances de cada uno de estos enfoques.

Enfoque Analítico	Enfoque Sistémico
Aisla: Se concentra sobre los elementos.	Relaciona: Se concentra sobre las interacciones de los elementos.
Considera la naturaleza de las interacciones.	Considera los efectos de las interacciones.
Se basa en la precisión de los detalles.	Se basa en la percepción global.
Modifica una variable a la vez.	Modifica simultáneamente grupos de variables.
Independiente de la duración: Los fenómenos considerados son reversibles.	Integra la duración y la reversibilidad.
La validación de los hechos se realiza por la prueba experimental en el marco de una teoría.	La validación de los hechos se realiza por comparación del funcionamiento del modelo con la realidad.
Modelos precisos y detallados, aunque difícilmente utilizables en la realidad.	Modelos insuficientemente rigurosos para servir de base a los conocimientos, pero utilizables en la decisión y en la acción.
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 pluri-disciplinaria.

Conduce a una acción programada en sus detalles.	Conduce a una acción por objetivos.
Conocimiento de los detalles, objetivos mal definidos.	Conocimiento de los objetivos, detalles borrosos.

En los últimos años el enfoque de sistemas encuentra cada vez mayor aplicación no sólo en la ciencia, sino que también en la tecnología. La construcción de un mecanismo complejo técnico moderno, como lo puede ser el sistema de dirección de un determinado proceso productivo, como un sistema digital de control distribuido en la industria de proceso continuo, por ejemplo, presupone un cálculo exacto de todo el conjunto de condiciones en que debe trabajar dicho mecanismo, y una rigurosa coordinación de los esfuerzos de toda una serie de instancias de soporte teórico conceptual. Es decir, para ello es preciso no sólo una representación espacial del mecanismo en sí, sino que también del proceso de su creación en el cual normalmente participan grupos de trabajo diferentes.

Por lo tanto, una de las causas que impulsaron la búsqueda de nuevos enfoques y métodos fue el carácter cada vez más limitado de las posibilidades heurísticas del elementalismo y del mecanicismo, y también las exigencias de la ciencia y de la tecnología, que cada vez más se fueron enfrentando a objetos y fenómenos cuya característica fundamental es su dimensión sistémica.

De todos modos, el enfoque sistémico no se puede considerar aún como un enfoque científico general, por cuanto hoy no se aplica en todas las ciencias, y a que muchos científicos opinan que enfoque de sistemas resulta inefectivo en las disciplinas científicas en las cuales ellos desenvuelven su quehacer, sobre todo en las más tradicionales y clásicas.

Sin embargo, y dadas las tendencias actuales y el hecho de que cada vez que los objetos de investigación aparezcan distinguidos como sistemas, es de esperar que muchas disciplinas científicas en las cuales el enfoque sistémico es poco efectivo y conducente a trivialidades, acepte este cuerpo de ideas o por lo menos algunas de ellas.

5. TEORÍA GENERAL DE SISTEMAS.

5.1 CONCEPTO DE SISTEMA.

En el dominio de las explicaciones científicas existen, según H. Maturana, dos caminos explicativos, los cuales se distinguen por el *hacerse cargo o no de la pregunta por el observador de los fenómenos*.



FIGURA 1.

El primer camino explicativo (lado izquierdo de la Fig. 1) asume que la realidad existe con independencia del observador. El problema del conocer se reduce a dar fiel cuenta de esa realidad. En este camino explicativo, al que Maturana denomina de las *Ontologías Trascendentales*, no se admite la pregunta por el observador. La realidad es una sola, es objetiva y universal. Los criterios de validación para el conocimiento descansan en categorías universales que trascienden al observador; materia, energía, mente, etc.

El segundo camino explicativo (lado derecho de la Fig. 2) establece que "la realidad sólo existe para un observador en el lenguaje". La realidad se configura producto de coherencias operacionales entre las operaciones de distinción que, en su operar, desarrolla un observador. A este camino Maturana lo denomina el de las *Ontologías Constitutivas*. En él surgen muchas realidades, cuyas fronteras se clausuran en las *coherencias operacionales* que un observador experimenta al concatenar, en base a ciertas premisas, las operaciones de distinción que configura. De este camino se desprende que los hombres existen en un multiverso, en múltiples dominios de acción, en múltiples dominios lingüísticos. La objetividad es, entonces, entre paréntesis, y en ella se manifiesta la ineludible vinculación entre "realidad" y observador: *éste puede ver sólo lo que es capaz de distinguir*.

En la perspectiva de las Ontologías Constitutivas, toda unidad sólo aparece cuando tras una *operación de distinción* es separada de un cierto trasfondo o sustrato, estableciendo sus límites, o, en los términos de H. Maturana, "Una distinción consiste en que un observador especifica lo que distingue al traerlo a la mano con lo que hace". De este modo, por ejemplo,

un trozo de ónix es un trozo de ónix si un observador lo distingue como tal, pero si otro observador lo distingue como un cenicero, entonces lo que se tiene es un cenicero.

Todo esto significa que es imposible tener noción de algo con independencia de la operación de distinción que lo distingue, es decir, sin operación de distinción ninguna cosa hay para el observador.

Por lo tanto, la existencia de un sistema, al igual que la existencia de cualquier objeto, dependerá de la existencia de un observador, que mediante una operación de distinción lo distingue como tal.

Aquella operación de distinción es la operación básica que cualquier observador ejecuta en su vivir cotidiano. Es por lo tanto en esta operación en la que se genera:

- Una unidad.
- El medio en que ella es distinguida.

Es decir, en la distinción señala la unidad, ejecutando la operación que define sus límites, separándola respecto de un trasfondo. Esta distinción involucra a todas las coherencias operacionales que hacen posible la distinción de la unidad.

En esta operación de distinción, el observador puede distinguir dos clases de unidades: simples y compuestas.

Unidades Simples.

Una *unidad simple*, al ser distinguida, corresponde a una entidad que queda caracterizada por un conjunto de propiedades. Esto significa que se constituye como un todo al especificar sus propiedades como una colección de dimensiones de interacción en el medio en que es distinguida.

Queda, por lo tanto, caracterizada por las propiedades a través de las cuales ella es generada, lo cual implica que no se requiera de ninguna explicación adicional sobre el origen de estas propiedades; es decir, se distingue algo sobre lo cual no se va a aplicar nuevamente operaciones de distinción. No existe, en este campo, operación de descomposición.

Cabe destacar que cuando se hace referencia a las propiedades de aquello que es distinguido como *unidad simple*, no se está haciendo referencia a características intrínsecas a la unidad. Se trata de propiedades a las cuales se hace referencia en la operación de distinción. Es decir, sólo desde el momento en que se hace la distinción se opera con la *unidad distinguida*. No se trata, por lo tanto, de propiedades independientes de la distinción del observador.

Unidades Compuestas.

Una *unidad compuesta* es una *unidad distinguida* en un primer momento como *unidad simple* pero que, a través de operaciones de distinción adicionales, es descompuesta por el observador en componentes tales que, a través de su composición, constituyen la *unidad simple original*, en el dominio en que ella es distinguida.

Por lo tanto, distinguir una unidad compuesta implica dos operaciones:

- Distinguir para generar una totalidad como unidad simple.
- Distinguir para tras una operación de descomposición, generar los componentes de la totalidad.

Esto significa que una unidad compuesta es distinguida en dos dominios: el dominio en que se distinguen los componentes, y el metadominio en que se la distingue en cuanto unidad simple.

Esto es así porque la totalidad o unidad simple aparece como resultado de una operación de composición.

Aquellas dos operaciones: distinguir una totalidad y descomponer son operaciones conexas, lo cual implica que los componentes de una unidad compuesta son componentes en la composición de la unidad compuesta.

Esto significa que al distinguir una totalidad y realizar sobre ella una operación de descomposición, ésta no puede ser cualquier descomposición. Debe ser tal que en la recomposición se genere la unidad compuesta como totalidad, con las propiedades que tiene como totalidad.

De esto se deduce que entre los componentes de una unidad compuesta y su correspondiente totalidad o unidad simple, existe una relación constitutiva de mutua especificación.

Por lo tanto, las propiedades de una unidad compuesta, distinguida como una unidad simple, que contiene las propiedades de las componentes que la constituyen como tal, y las propiedades de las componentes de una unidad compuesta y su manera de composición, determinan las propiedades que la caracterizan como unidad simple, cuando es distinguida como tal.

Esto significa que los componentes de una unidad compuesta lo son exclusivamente en relación a la composición de la unidad simple que ellos conforman.

Aún más, cuando se distingue una unidad compuesta, se está implicando una determinada composición y no otra. La composición que resulta en que lo distinguido sea la unidad simple distinguida.

En estos términos, entonces, no es correcto distinguir un componente independientemente de la unidad que integra. Así como no es correcto descomponer la unidad simple, al distinguirla como unidad compuesta, en un conjunto arbitrario de componentes y conformarlos en una forma arbitraria de composición.

Por lo tanto, si, tras distinguir los componentes en una unidad simple, que se trata como unidad compuesta, aquellos se agrupan de manera que no configuran la unidad original, se está descomponiendo una unidad diferente de la unidad inicial, y los componentes distinguidos no son componentes de la unidad que el observador asumía que ellos componían.

Unidades Compuestas y Sistemas.

Toda vez que un observador distingue una unidad compuesta, lo que distingue es un sistema. Es decir, aparece un sistema cuando una unidad se distingue como totalidad, y posteriormente se distinguen elementos como componentes de ella, dispuestos de una manera tal que, como resultado de una operación de composición, se obtiene la unidad que el observador distinguió inicialmente como totalidad, como aparece en la Fig. 2.

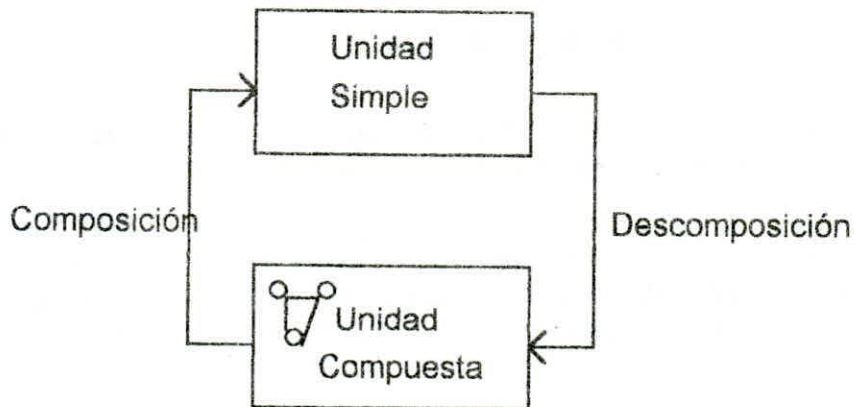


FIGURA 2

Por ejemplo, si una familia (*familia X*) es distinguida como sistema, esto implica que la condición *familia X* es una unidad simple que se distingue inicialmente, y la referencia a los componentes en una distinción posterior, distingue la unidad compuesta, momento en el cual la *familia X* aparece como sistema, toda vez que sus miembros configuran la unidad simple *familia X*.

La *familia X*, como unidad simple, existe en un dominio distinto de aquel configurado por la *familia X* en cuanto unidad compuesta, y se puede interactuar con ella en ambas dimensiones.

Si *familia X* se registra en una solicitud, por ejemplo, se interactúa con la unidad simple. Pero si se visita a la *familia X*, se interactúa con ella en cuanto unidad compuesta, ya que se está, en este caso, interactuando con los componentes.

Por lo tanto, operacionalmente, un sistema es cualquier conjunto interconectado de elementos que un observador puede, de alguna manera, tratar como una totalidad.

Esto significa que se distingue un sistema cada vez que se distingue un conjunto de elementos tales que las relaciones entre ellos especifican un borde operacional con respecto a cualquier otra cosa, o con respecto a un dominio cualquiera.

De este modo, entonces, si un observador no logra mostrar que las relaciones entre los elementos son preferenciales entre ellos, de modo que se establezca un borde con respecto a otros elementos, no se puede hablar de un sistema.

5.2 ORGANIZACIÓN Y ESTRUCTURA.

En todo sistema el distinguir una totalidad y hacer una distinción adicional de los componentes que en sus interacciones constituyen aquella totalidad, que es esa totalidad y no otra, aparecen dos dimensiones: la *organización* que confiere su identidad al sistema, y la *estructura* que materializa la organización.

Organización.

La organización de una unidad compuesta corresponde a la configuración de las relaciones, estáticas o dinámicas, entre los componentes, que especifican su identidad de clase como unidad compuesta, la que puede ser distinguida como unidad simple de una clase particular.

Es decir, la organización de una unidad compuesta se constituye cuando las relaciones entre componentes, que la hacen unidad compuesta de una cierta clase, especifican su identidad de clase como unidad simple en un metadominio con respecto a sus componentes.

Por lo tanto, si cambia la organización de la unidad compuesta, la unidad pierde su identidad de clase, desintegrándose en cuanto sistema.

Esto significa que la organización de una unidad compuesta es necesariamente invariante, mientras conserva su identidad de clase, y ésta se conserva mientras se mantiene la organización de la unidad compuesta.

Estructura.

En una unidad compuesta, sea estática o dinámica, la estructura queda constituida por los componentes actuales y las correspondientes relaciones entre ellos, en tanto realizan, en cada momento, la organización de la unidad compuesta.

Es decir, al hablar de estructura de una unidad compuesta se está haciendo referencia a los componentes más las relaciones entre ellos, los cuales configuran un determinado sistema y no otro. Por lo tanto, puede decirse que la diferencia fundamental entre estructura y organización radica en que la primera hace referencia a los componentes, tal como aparece en la Fig. 3.



FIGURA 3.

Intersección Estructural.

Aquellos elementos que son componentes de un sistema pueden ser componentes de otro sistema simultáneamente, o sucesivamente, sin dejar de ser componentes de un sistema particular en un momento determinado.

Esto significa que si algunos componentes de un sistema S1 son componentes de un sistema S2, los sistemas S1 y S2 se intersectan en la estructura, pero no en la organización.

Alcances en lo Organizacional.

Preguntar por la organización de una unidad determinada implica hacer referencia a las relaciones que definen la identidad de clase de la unidad compuesta que se observa. No es lo mismo decir "cuál es la organización de la *familia*" a decir "cuál es la organización de la *familia X*".

Esto significa, en esencia, que una organización siempre se connota con una operación de distinción, e implica que todas las unidades compuestas que, por ejemplo, satisfacen la organización de la *familia X*, serán *familia X*.

Cuando se dice "este es un computador", implicando con ello una organización, en el dominio de las unidades compuestas, implica una organización, pero si se dice "este es un computador Apple" se implica otra organización. En este caso, se implica las relaciones que definen el computador Apple como computador más las relaciones que lo definen como computador Apple.

Si lo que se distingue es "computador", hay una serie de relaciones que pasan a ser parte de la estructura, porque son irrelevantes con respecto a la organización "computador Apple".

Del mismo modo, al distinguir "computador Apple", se está introduciendo en la organización ciertas relaciones que al distinguir "computador" pertenecen a su estructura.

5.3 VARIABILIDAD E INVARIANZA DE UN SISTEMA.

Cuando se reconoce que la organización define un sistema, éste aparece en el instante en que se constituye la organización y se conserva mientras ella se conserva, pero desaparece en el momento en que esa organización deja de conservarse. Esto significa que todo sistema es constitutivamente conservador.

Sin embargo, la estructura puede variar, ya que corresponde al modo particular en que se realizan las relaciones de la unidad compuesta. Por lo tanto, al observar el comportamiento estructural del sistema, se puede observar la dinámica de cambio de la red sistémica, e identificar un *espacio de variación* mientras se conserva la organización. Por lo tanto, en el momento en que se identifica un conjunto de elementos constituyendo un sistema, se especifica su organización y su *dominio de variabilidad*.

Un sistema cuya estructura puede cambiar mientras su organización permanece invariante es una unidad plástica, y por lo tanto un sistema dinámico. Además, las interacciones del sistema, en el espacio de sus componentes, son interacciones estructurales, y las

interacciones bajo las cuales se mantiene aquella invarianza se reconocen como *perturbaciones*.

Toda variación de un sistema, en tanto mantiene su organización, es decir, todo cambio de estructura sin pérdida de la identidad de clase, corresponde a los *cambios de estado* del sistema.

Las perturbaciones son eventos activantes que acoplan la secuencia del cambio de estado de la unidad compuesta a la secuencia de los cambios de estado del medio que constituyen las perturbaciones.

Dado que la mantención de la organización de un sistema, bajo condiciones de cambio estructural, es un rasgo constitutivo de estas unidades compuestas, la realización de la organización de una unidad compuesta plástica, sometida a perturbaciones generadas por un medio cambiante, resulta en el establecimiento en el sistema de una estructura que puede generar cambios específicos de estado que se pueden provocar por cambios perturbadores específicos del estado del medio.

Esta correspondencia estructural dinámica se conoce como *acoplamiento estructural* y puede describirse como la correspondencia efectiva espacio-temporal de los cambios de estado del sistema con los cambios de estado recurrentes del medio, mientras el sistema mantiene su organización.

Esto significa que la estructura del sistema determina qué admite como perturbación. Es decir, es la estructura del sistema lo que determina qué configuración estructural admite en el encuentro.

La ontología, la historia individual de un sistema, necesariamente transcurre bajo condiciones de conservación de su correspondencia con el medio, en donde ésta implica el encontrarse sólo bajo perturbaciones, ya que con la primera interacción destructiva el sistema se destruye. Esta correspondencia con el medio se denomina *adaptación*.

Obviamente, las interacciones destructivas las determina el sistema a cada instante.

Por lo tanto, la historia individual de un sistema transcurre necesariamente bajo dos condiciones invariantes: conservación de organización y adaptación. De esta manera todo sistema tiene la configuración estructural que tiene, en la circunstancia actual, como resultado de una *ontogenia*⁴ con conservación y adaptación.

Cuando la ontogenia transcurre bajo condiciones de conservación de organización y adaptación, se está frente a interacciones recurrentes entre sistema y medio, en donde el sistema se desliza en congruencia con aquel medio, configurando un cuadro en donde sistema y medio van cambiando juntos.

Como sistema y medio cambian juntos, existe congruencia entre la historia de cambio estructural del sistema y la del medio. El curso que se gesta, momento a momento, en las interacciones sistema-medio, se denomina *deriva*. Por lo tanto, la historia de cambio estructural de un sistema en interacciones con el medio, es una *deriva estructural*.

⁴Ontogenia: Experiencia.

Si las interacciones recurrentes ocurren entre dos seres vivos, entre ellos se producen cambios congruentes o se separan. Por lo tanto, si interactúan recurrentemente, cambian juntos.

Si se observa esta situación desde la perspectiva de un observador, lo que se ve son *coordinaciones conductuales*, que son coordinaciones de acción, cabe destacar que el observador no ve que en la medida en que estén en interacciones recurrentes, las formas de esas coordinaciones de acción van cambiando a lo largo de la historia de interacciones.

El observador ve las coordinaciones conductuales que se establecen en la historia de vivir juntos, coordinaciones que se denotan como *consenso*, término que se emplea para hacer referencia a todas las coordinaciones conductuales que se establecen como resultado de la historia de vivir juntos sin ser explicitadas.

Por lo tanto, cuando la estructura de dos seres vivos, percibidos como sistemas, en interacciones recurrentes cambian como resultado de esas interacciones recurrentes, lo que ve un observador son *coordinaciones consensuales*, es decir, observa un dominio de coordinaciones conductuales consensuales.

Si se satisfacen ciertas circunstancias en lo que se refiere a la diversidad interna posible y la historia de interacciones recurrentes, en el sentido que ésta sea suficientemente intensa, puede ocurrir que el observador se encuentre en condiciones de decir que no solamente hay *coordinaciones conductuales consensuales*, sino que hay *coordinaciones conductuales de coordinaciones conductuales consensuales*, y en ese instante habría surgido el *lenguaje*. En este sentido, el lenguaje no requiere nada especial, sólo interacciones recurrentes. Y allí todas las operaciones que se pueden hacer en el lenguaje se hacen posibles. En el momento en que surge el lenguaje, surgen la simbolización, los objetos, el observador, la autoconciencia.

5.4 EL CARÁCTER ESTRUCTO-DETERMINADO DE UN SISTEMA.

La estructura de un sistema consiste en sus componentes y en las relaciones entre ellos. Dado que todo sistema es tal, en la medida que mantiene su organización, todo cambio en una unidad compuesta corresponde necesariamente a un cambio estructural, el que ocurre determinado por la estructura del sistema, como consecuencia de la operación de las propiedades de sus componentes.

Del mismo modo, cuando el cambio estructural de un sistema se da como resultado de una interacción, también aparece determinado por su estructura, ya que aquel cambio estructural tiene lugar al interior de las propiedades de la unidad compuesta, tal como ellas están involucradas en su composición. Esto significa que un agente externo, que interactúa con el sistema, sólo puede *gatillarle* cambios, no determinárselos. Más específicamente, nada externo al sistema puede especificar lo que en ellos pasa. No hay interacciones instructivas. Es decir, la determinación estructural implica que en un sistema existe un conjunto de elementos conectados de una manera tal, que lo que pasa con ese conjunto de elementos depende de cómo está hecho.

Esta condición implica, además, que la estructura de una unidad compuesta determina con qué configuración estructural del medio ella puede interactuar.

H. Maturana ejemplifica esta determinación estructural cuando al respecto dice: "si yo me encuentro con una grabadora, aprieto un botón esperando que la grabadora empiece a funcionar y no funciona, no voy al médico para decirle: doctor, examíneme el dedo, que mi grabadora no funciona cuando aprieto el botón. No hago eso, agrega, porque no es el dedo el que determina lo que le pasa a la grabadora, sino que el dedo gatilla lo que le pasa a la grabadora".

5.5 LAS POSIBILIDADES DE INTERACCIONES EN UN SISTEMA.

Cuando se distingue una unidad simple, al momento de la distinción se especifican sus propiedades en términos de las operaciones de separación que se hacen para distinguirlos. Por lo tanto, la interacción es en términos de sus propiedades distinguidas.

Cuando se trata con un sistema, al descomponer la totalidad distinguida en un comienzo como unidad simple, aparecen sus componentes, con los cuales se interactúa, por su parte, en dos dimensiones. Los componentes son componentes en cuanto a la satisfacción de las relaciones de composición, por lo cual expresan relaciones relativas a la composición del sistema, pero también expresan otras relaciones, más allá de las relativas a la composición.

Esto significa que se puede interactuar:

- En las dimensiones de composición de los componentes, en donde lo que se gatilla como cambio estructural es un cambio que se gatilla en la dinámica del sistema, razón por la cual si se interactúa con un componente, lo que le ocurre al sistema es lo mismo que le ocurre si interactúa con otro componente. En este tipo de interacción aparece como lo más relevante la conformación del sistema.
- En las dimensiones no sistémicas de los componentes, en este caso, lo que se gatilla son cambios estructurales en los componentes del sistema, los cuales gatillan cambios en las características del componente. Estos cambios pueden ser irrelevantes para la realización de las dimensiones de constitución del sistema, es decir, no tener consecuencias en su constitución, al gatillar cambios estructurales en los componentes que pueden ser no conformadores del sistema.

Cuando ocurren interacciones del segundo tipo, H. Maturana habla de interacciones *ortogonales*, porque se expresan en una dirección distinta de la dirección de las relaciones de composición.

Cabe destacar, no obstante, que cada vez que hay interacción, hay gatillamiento de cambios estructurales. Lo que puede ocurrir es que se gatillen cambios estructurales irrelevantes para la dimensión sistémica, lo cual significa que hay *interacciones relevantes e interacciones irrelevantes* para la conformación del sistema.

5.6 EL CARÁCTER CERRADO DE UN SISTEMA.

En todo sistema las interacciones que éste experimenta en su relación con el medio, dado su carácter estructo-determinado, no especifican lo que le sucede en cuanto a sistema.

Todo sistema queda definido por su organización. Ésta, por su parte, se realiza a través de un conjunto de componentes que satisfacen sus relaciones de composición. En esta

perspectiva el sistema es *cerrado*. Sin embargo, desde el punto de vista de su estructura, si se trata de un sistema material, es *necesariamente abierto*. Cabe destacar, en todo caso, que esto no significa que el sistema sea en algunos aspectos abierto y en otros cerrado. El sistema, en términos de aquello que lo define como sistema, su organización, es cerrado.

Lo que ocurre es que al hablar del carácter cerrado del sistema se está hablando de organización, y al hablar de la apertura del sistema, se está haciendo referencia a la estructura.

Un organismo, por ejemplo, en esta perspectiva, es abierto, porque de lo contrario no se tienen las moléculas que requiere. Es decir, los sistemas, para, en último término, conservar su organización, deben constantemente transformar la energía y emplearla según sus conveniencias. Esto es equivalente a hablar de autoconservación, ya que mantener la organización requiere invariablemente la energía.

En síntesis, esto significa que si se pone énfasis en la organización del sistema, en aquello que lo define, se trata de un sistema cerrado. Pero, como lo dice Bertalanffy, si se pone énfasis en los cambios energéticos, es abierto. Claro está que en este caso no se está observando el sistema en relación a aquello que lo define.

5.7 SISTEMAS Y SUBSISTEMAS.

En un sistema los componentes aparecen cuando, tras distinguir en un primer momento una unidad simple, se realiza una operación de distinción adicional. Tienen la categoría de componentes cuando al ejercer sobre ellos una operación de composición, constituyen la unidad simple original.

Si estos componentes pueden, a su vez, distinguirse como unidades compuestas, aparecen entonces como sistemas, los cuales, en el contexto del sistema original, son susceptibles de considerar como subsistemas.

Si esta operación de distinguir sucesivas unidades compuestas, o sistemas, a partir de unidades compuestas distinguidas previamente, puede realizarse recurrentemente, aparece entonces el sistema originalmente distinguido desplegado de manera vertical.

Aquella verticalidad se expresa en términos de *estratos* y *niveles sistémicos*, donde cada uno de estos niveles corresponde a un subsistema de la unidad compuesta distinguida en un comienzo, y cada estrato a los dominios de interacción que aparecen tras las distinciones realizadas recurrentemente a partir de la distinción del sistema original.

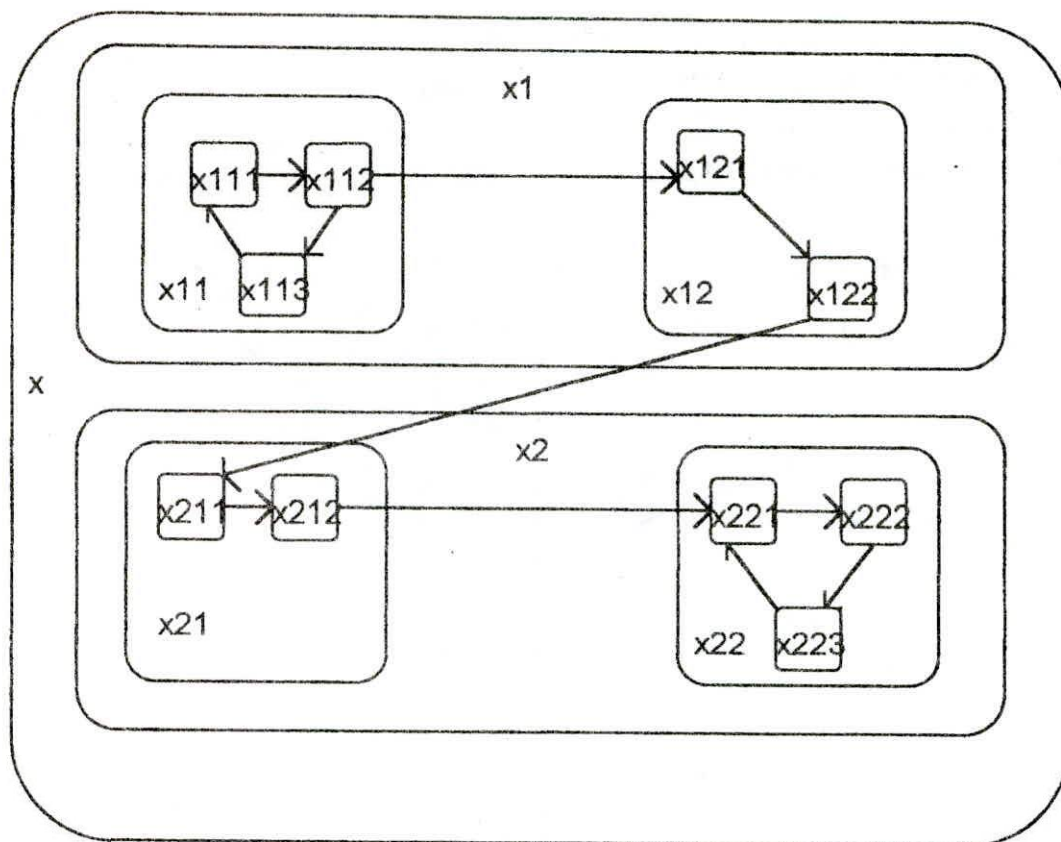


FIGURA 4

En la Fig. 4 aparecen cuatro estratos y tres niveles sistémicos. Tres niveles sistémicos implica tres subsistemas, ya que la configuración corresponde a la distinción recurrente de tres unidades compuestas, y cuatro estratos, ya que los estratos que aparecen en la verticalidad generada por las distinciones realizadas a partir de un sistema distinguido originalmente corresponden a:

$$e = ns + 1$$

donde: e = estratos

ns = niveles sistémicos

Cabe destacar que cada uno de aquellos subsistemas, al ser distinguido como unidad compuesta, en su propio nivel sistémico, corresponde a un sistema, cuando son considerados como una primera unidad compuesta inicialmente distinguida.

Estructuras Sistémicas Multiniveles.

Una de las características de los sistemas viables, asociada a la configuración en niveles sistémicos, es la tendencia a formar estructuras de múltiples niveles, de acuerdo a un padrón organizacional dado, y que difieren en su complejidad. Esta tendencia, incluso, puede ser vista como un principio básico de autoorganización.

En cada nivel de complejidad aparecen sistemas integrados, que consisten en partes menores y al mismo tiempo actúan como partes de totalidades mayores.

Conforme a Capra, las relaciones entre esos niveles sistémicos puede representarse por un *árbol sistémico*. En él existen interligazones e interdependencias entre todos los niveles.

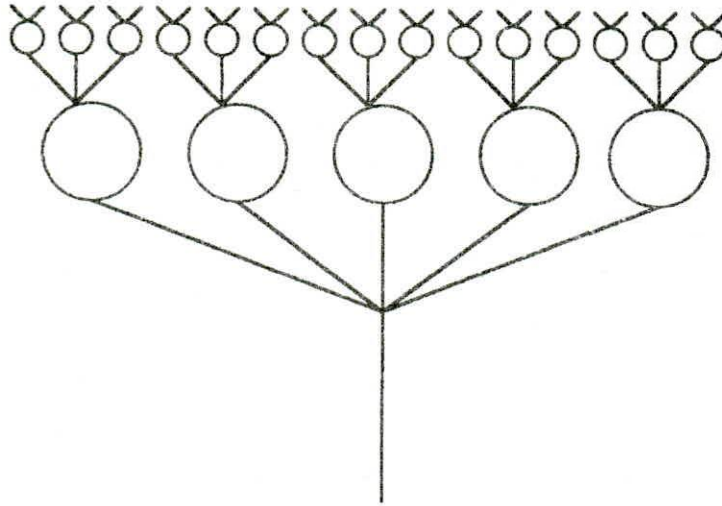


FIGURA 5.

Los diferentes niveles de complejidad de un sistema vivo corresponden a aquellos que aparecen en la Fig. 5. En esta configuración, en los sistemas vivos, por ejemplo, en cada nivel el sistema que está siendo considerado puede constituir un organismo individual. Una célula puede ser parte de un tejido, pero también puede ser un microorganismo, que a su vez es parte de un ecosistema.

Todo subsistema es un organismo relativamente autónomo, pero al mismo tiempo un componente de un organismo mayor. Manifiesta las propiedades independientes del todo, y las propiedades independientes de las partes. De este modo, entonces, el orden en un nivel sistémico es la consecuencia de la autoorganización en un nivel mayor.

Estas configuraciones estratificadas permiten a los sistemas vivos evolucionar más rápidamente, y les otorgan una probabilidad mayor de sobrevivencia, ya que en este caso, y frente a graves perturbaciones, pueden descomponerse en sus diferentes subsistemas sin ser completamente destruidos. Frente a esta situación, los sistemas no estratificados desaparecerían totalmente y tendrían que comenzar a evolucionar de nuevo.

Cabe destacar que la estratificación de los sistemas viables es diferente de la noción de jerarquía. El padrón organizacional *jerárquico* hace referencia a escalones de acuerdo a niveles de poder, estando cada escalón subordinado a otro de nivel superior.

De acuerdo a esta perspectiva interpretativa, en el pasado, el orden estratificado de la naturaleza fue frecuentemente mal interpretado, con el propósito de justificar las estructuras políticas y sociales autoritarias.

En el orden estratificado de los sistemas, lo central no consiste en la transferencia de control, sino en la organización de la complejidad. De este modo, los niveles sistémicos son niveles estables de diferentes complejidades, lo que da la posibilidad de, en la distinción, usar descripciones diferentes para cada nivel.

Por cierto que cualquier nivel que se distingue corresponde realmente al nivel de atención del observador.

5.8 AUTONOMÍA.

La *autonomía* es el rasgo fundamental que caracteriza a los sistemas. Ésta queda de manifiesto claramente en los sistemas viables, particularmente en los sistemas vivos. Hablar de autonomía implica referirse a autoproducción, razón por la cual los sistemas viables sólo se caracterizan adecuadamente como una red de procesos de producción de componentes que son continua y recursivamente generados y realizados como una entidad concreta (unidad) en el espacio físico, a través de las interacciones de los mismos componentes que los producen como tal red. Maturana llama a esta organización una *organización autopiética*; por lo tanto, cualquier sistema que exhiba aquella organización es un *sistema autopiético* en el espacio en que existen sus componentes.

La autonomía de los sistemas viables, producto de su organización autopiética, operando a través de una estructura internamente determinada, permite poner de manifiesto el carácter de autoorganizados de estos sistemas.

5.9 LAS TOTALIDADES EMERGENTES.

Un aspecto fundamental que aparece cuando se distingue un sistema, es que ellos poseen atributos diferentes a los elementos que lo constituyen. Del mismo modo, poseen modos de acción propios, los cuales no se derivan solamente de los modos de acción de los elementos que lo constituyen.

Puede decirse que aparece un tipo de *rol organizador* del todo en relación con sus componentes. Esta *emergencia* del todo puede observarse en los más diversos fenómenos. Se tiene como ejemplos los compuestos químicos en contraposición a los agregados físicos de elementos, la materia viva en comparación con sistemas fisicoquímicos, los organismos animales frente a las células individuales de materia viva, los procesos físicos en relación a las propiedades biológicas de un organismo, y las sociedades humanas en relación a las propiedades biofísicas y modos de conducta del organismo humano.

Esto significa que al referenciar un sistema, tras los procesos de distinción correspondientes, se hace necesario fundamentarlo en *lo que no es sistémico*, o más específicamente, en lo que no ha de ser analizado como tal y constituye el límite al cual ha de llevarse el análisis, es decir, lo contrario al sistema, pero que lo constituye, como lo es el *elemento*.

Así, si se explicita un determinado sistema, lo que se hace es que, a la par de haberlo determinado, se ha fijado frente a él ciertos elementos cuyas relaciones, propiedades, e interacciones forman el sistema, aunque ellos en sí mismos no lo son.

Si por otra parte se presta atención a los elementos como tales, considerándolos a su vez como sistemas por separado, se está ahora fuera de los marcos teóricos en el cual dichos elementos actúan en calidad de tales, y se pasa a otro sistema y por ende a otro nivel de análisis, el cual no necesariamente debe tener las mismas características que el anterior.

Por lo tanto, *el sistema es una unidad integral*, que se expresa mediante el concepto de *totalidad*. Entonces, hablar de un sistema, entendido como totalidad, es querer decir que se está frente no solamente a un conjunto, sino que a un conjunto de elementos interconectados, que logra su integridad debido justamente a dichas interconexiones.

El concepto de totalidad en este caso cumple la doble función de indicar que el sistema es una unidad de los aspectos contradictorios, como lo son la *separación* expresada por la pluralidad de elementos, y la *conexión* expresada como la unidad de esos elementos que constituyen un todo mayor, es decir, un sistema.

5.10 FENÓMENOS NO SISTÉMICOS.

No todos los objetos o fenómenos susceptibles de observar manifiestan características sistémicas. Un montón de piedras, las personas que caminan por la calle, no poseen características esenciales de organización interna. En estos casos las relaciones existentes son inesenciales, causales, y de carácter externo.

Si esos *conglomerados* se consideran conjuntos, los componentes, al entrar o salir de ellos, no experimentan cambio alguno, lo que deja de manifiesto la ausencia de propiedades integrales en esos conjuntos. Más aún, en tales conjuntos las propiedades del todo son coincidentes esencialmente con la suma de las propiedades consideradas individualmente.

Pero incluso cuando existen relaciones internas entre los componentes, ello no implica necesariamente la existencia de aquel rol organizador del todo. Dos moléculas de gas chocando entre ellas manifiestan una interacción mecánica, pero en este caso se trata de un simple sumativo, ya que como resultado de su interacción no surgen en él propiedades integrativas totales completamente nuevas.

5.11 HOMEOSTASIS.

La mantención de la organización, en un ambiente de cambio estructural constante, producto de interacciones gatilladas por el medio, exigen de los componentes estructurales un comportamiento en términos de procesos capaces de realizar las operaciones que la mantención de la organización y adaptación requiere.

Para ello es preciso que estos procesos mantengan su capacidad fisiológica en condiciones de realizar las tareas operacionales que la generación continua de los componentes requiere, a objeto de permitir la necesaria mantención de la organización. Para que esto ocurra, los procesos, configurados a partir de los componentes del sistema, asumen un comportamiento homeostático, al igual que el sistema, más específicamente, su organización, en cuanto totalidad de una determinada clase.



fisiológicos que el organismo no pueda soportar. En el organismo humano no sólo la temperatura de la sangre tiene que mantenerse en equilibrio, sino que también todo el sistema biológico del cuerpo. Para esto, los sistemas vivos disponen de un conjunto de mecanismos homeostáticos, mecanismos de regulación y de control que le permiten enfrentar las amenazas a su viabilidad que proviene de su entorno.

En este contexto resulta interesante destacar lo que apunta J. de Rosnay en relación a los sistemas fisiológicos; éstos, como lo señala de Rosnay "... muestra que las reacciones del hombre y de los animales a estas agresiones se reducen a tres comportamientos básicos: la adaptación, la fuga y la lucha".

Si el entorno, agrega de Rosnay, se hace desagradable, hostil o peligroso, el organismo puede reaccionar por la fuga: simplemente cambia de entorno, hasta asentarse en un medio donde se sienta a gusto. También puede atacar, defenderse, o modificar por una acción deliberada el entorno que le amenaza y volver a condiciones que le son favorables.

De esta forma el organismo se adapta constantemente a nuevas condiciones de vida, aunque ello no siempre es exactamente riguroso, ya que si se observa a un ser humano en proceso de adaptación, se puede apreciar que normalmente experimenta dificultades, lo que le provoca desasosiegos y frustraciones, aunque con toda seguridad estos tienen, en última instancia, un sentido positivo por cuanto son el punto de partida de las transformaciones o cambios.

Por otro lado, un ser humano afectado por su entorno se condiciona para la acción. Su organismo, señala al respecto de Rosnay, "moviliza reservas energéticas y segrega determinadas hormonas, como la adrenalina, que le prepara a huir o a luchar. Esta movilización se expresa por manifestaciones fisiológicas bien conocidas". Por ejemplo, es constatable el hecho de que el corazón late más rápidamente frente a un peligro, un mayor esfuerzo, o simplemente una emoción cualquiera, a la par que cambia el ritmo de la respiración y el cuerpo transpira, entre otras manifestaciones orgánicas.

Todas estas son manifestaciones fisiológicas, en virtud de las cuales se expresa la actividad del organismo, tiende a la mantención de su equilibrio interno, actividad que puede ser involuntaria como ocurre cuando se transpira, o bien voluntaria como beber cuando se tiene sed.

En otras palabras, la homeostasis es aquella característica de los sistemas viables, que permite que el equilibrio oscile en todo instante debido a las condiciones cambiantes del medio externo, ya sea, en todo momento, reestablecida por ciertos mecanismos reaccionales, como lo son los mecanismos de regulación y control de que dispone aquel sistema.

Dadas, entonces, las condiciones de la existencia de un sistema viable, la homeostasis, a través de los mecanismos de regulación y control, genera una conducta, cuya manifestaciones primordiales tienden a que en los sistemas se produzca la conservación del equilibrio, entre el medio externo que lo rodea y su medio interno.

El medio externo está compuesto por un número infinito de variables que hacen que su variedad⁵ sea superior a la que manifiesta el sistema. El sistema, al tener menor variedad,

⁵Variedad: Conjunto de estados posibles.

tiende a ser relativamente uniforme, en relación al permanente cambio que muestra el entorno. Por esta razón, es posible afirmar que aquel sistema debe adecuarse al primero.

Se puede decir, que si un estado de equilibrio permanente trae como consecuencia variaciones fisiológicas del sistema, éstas pueden ser tales que, dependiendo de la duración de dicha variación, generen cambios cuya mayor o menor permanencia lleven aparejados un comportamiento adaptativo. De esta manera entonces, el sistema puede enfrentar una perspectiva evolutiva.

En los sistemas vivos, esta evolución se da por medio de adquisiciones viables y colectivas, que al transmitirse por herencia o repetirse por persistencia de los factores causales, llevan consigo las transformaciones de aquel sistema vivo.

En este contexto, existe lo que se puede denominar una relación directa entre las transformaciones del sistema viable y los cambios en el medio, puesto que en la medida que se producen estas transformaciones, se produce una nueva actividad en el sistema, por lo cual, a su vez, sus acciones que evolucionan concretamente sobre el entorno, también se modifican. Por lo tanto, se puede decir que los cambios del medio generan cambios estructurales con mantención de organización en el sistema, y el sistema resultante, cambios que se expresan directamente por los cambios en su actividad, los cuales por su parte, transforman a su vez al medio. Es de este modo, entonces, que se genera el acoplamiento estructural sistema-medio.

Pero si el equilibrio se convierte en permanente, se produce la igualación energética de la entropía. Si el sistema por esta razón se detiene en sus transformaciones, se convierte en un sistema evolutivo concluido, corriendo el riesgo de desaparecer por su incapacidad orgánica de producir variaciones reactivas, como respuesta a los estímulos del medio.

Es por eso que al hablar de los sistemas viables, es necesario considerar la dinámica de estos sistemas, la cual expresa el continuo cambio del sistema. Este cambio lleva a una continua renovación de las variables que conforman un sistema, no obstante que permanecen idénticas a ellas mismas. Surge así lo aparentemente paradójico de la dinámica de la mantención. Cabe destacar que la persistencia de esta dinámica lleva a lo que se puede denominar la estabilidad dinámica del sistema.

El biólogo W. Ross Ashby en "Proyecto para un Cerebro", destaca dos ideas relacionadas con la homeostasis: el concepto de equilibrio dinámico, y el concepto de adaptación. Pone en relieve que la homeostasis muestra:

- Que todo mecanismo está adaptado a su finalidad.
- Que ésta consiste en mantener los valores de ciertas variables esenciales, dentro de ciertos límites fisiológicos.

Ashby señala, además, que la casi totalidad del comportamiento del sistema vegetativo de los animales se debe a tales mecanismos.

Puede decirse que el equilibrio dinámico expuesto por Ashby, equivale a un equilibrio de flujos. Este resulta del ajuste de las velocidades de dos o más flujos que atraviesan un elemento de medida. El equilibrio dinámico se diferencia de un simple equilibrio estático, en que aquél puede ser adaptado, modificado y modulado permanentemente gracias a reajustes, a veces imperceptibles, en función de las perturbaciones o circunstancias, generándose así

los estados denominados *estacionarios*. Los estados dinámicos corresponden a la mantención constante de un cierto nivel, tasa, o parámetro en general, a pesar del cambio continuo en el cual están insertos, y es esto precisamente lo que lo diferencia de los estados estáticos, los cuales son también estables, pero en la inmovilidad.

Dicha dinámica, como expresa de Rosnay, lleva al sistema a tener múltiples estados estacionarios, permitiendo así que las estructuras de los sistemas puedan adaptarse y responder a la inmensa variedad de modificaciones del entorno, que se traducen en perturbaciones para el sistema, manteniendo su organización.

Con respecto a la adaptación, Ashby señala que "una forma de comportamiento es adaptativa, si mantiene las variables esenciales dentro de los límites fisiológicos", y como recalca J. Mélese, para que un sistema pueda sobrevivir en un contexto cambiante, debe ser capaz de evolucionar, es decir adaptarse a las modificaciones del universo que lo rodea.

Es así entonces que puede concluirse que los sistemas viables disponen de ciertos mecanismos que le permiten hacer frente a las perturbaciones del entorno, manteniendo el equilibrio del medio interno por medio de la conservación, en su valor normal, de las diferentes variables fisiológicas que los determinan. Los sistemas en los cuales esto ocurre, se denominan sistemas homeostáticos, y los mecanismos que le permiten alcanzar tal categoría son los mecanismos de regulación y control.

En consecuencia, un sistema homeostático, sea este una célula o una organización social, es un sistema cerrado que conserva su organización por medio de una multiplicidad de equilibrios dinámicos que se producen a través del operar estructural del sistema. Como señala J. De Rosnay, "equilibrios rigurosamente controlados por mecanismos de regulación interdependientes" y agrega "un sistema así configurado reacciona a todo cambio proveniente del medio, o a toda perturbación aleatoria, por una serie de modificaciones del igual valor y de dirección opuesta a las que lo generan: estas modificaciones persiguen la mantención de los equilibrios internos".

Los sistemas ecológicos, biológicos, y sociales, son particularmente homeostáticos. Como dice de Rosnay "se oponen al cambio por todos los medios a su disposición. Si el sistema no consigue restablecer sus equilibrios, entra entonces en otro modo de funcionamiento con restricciones cada vez más draconianas que las precedentes, y que pueden llevar, si continúan las perturbaciones, a la destrucción del conjunto".

Las propiedades autorreguladoras de estos sistemas fueron descubiertas por el cibernético W. R. Ashby. Para él, la homeostasis es el tipo de autocontrol que opera en los sistemas homeostáticos. En este caso el control está asociado a la retroalimentación, esto es, ocurre cuando una función de salida de un subsistema es parte de la entrada de otro, en este caso la salida del sistema controlado es la entrada del sistema de control.

5.12 HOMEOSTASIS Y ULTRAESTABILIDAD

De acuerdo a lo expuesto, puede decirse, entonces, que una de las propiedades básicas de la homeostasis es que permite que el sistema modifique sus estructuras, y por lo tanto sus relaciones externas, para mantener la constancia de una serie de indicadores fundamentales del medio interno.

La homeostasis se presenta como una condición esencial de la estabilidad, y por lo tanto de la sobrevivencia de los sistemas viables.

La estabilidad se puede definir en función de las perturbaciones del sistema. Cuando se perturba un sistema que funciona fluidamente, se perturba su rendimiento. Un sistema estable reestablece rápidamente esa producción en su valor anterior amortiguando la oscilación producida. Pero, obviamente, cuando se habla de reguladores de retroalimentación directa, es preciso conocer el tipo de perturbación. En el caso de un sistema diseñado por el hombre, puede permitir hacer frente a perturbaciones originadas en un número limitado de entradas conocidas, pero no es posible que ese sistema resuelva perturbaciones para las cuales no está diseñado. Obviamente un sistema viable debe superar aquellas limitaciones.

Es deseable que los mecanismos homeostáticos permitan realcanzar el equilibrio del sistema después de haber sido perturbado por cualquier causa, incluso por alguna nunca experimentada hasta ahora. Cuando eso es posible, se habla de *ultrastabilidad*, es decir, de la capacidad de un sistema de recuperar el equilibrio después de cualquier perturbación, incluyendo aquellas que el "diseñador" aún no ha considerado. Esto permite a Ashby hablar de *retroalimentación de segundo orden*, cuando se refiere a ultraestabilidad.

Por lo tanto, los sistemas homeostáticos son ultraestables, lo que significa que toda su estructura contribuye a la mantención de la organización que materializa.

De acuerdo a Ashby, todo sistema que incorpore retroalimentación negativa presenta un equilibrio estable y, a la inversa, todo sistema de equilibrio estable debe basarse en la retroalimentación negativa. Pero todo sistema homeostático presenta algo más que la estabilidad común, es ultraestable.

5.13 LA HOMEOSTASIS Y LOS LIMITES FISIOLÓGICOS.

Los sistemas vivos y los sistemas ecológicos, en general, presentan dos formas de autorregulación; la mantención de las variables críticas del sistema estables dentro de límites fisiológicos y la ultraestabilidad. Es preciso, entretanto, tener presente que las pretensiones de la homeostasis no son ilimitadas.

Los mecanismos homeostáticos mantienen las variables críticas del sistema estables dentro de límites fisiológicos. Esto significa que los límites son fijados por la capacidad del sistema para funcionar satisfactoriamente dentro de los mismos. Además, los límites fisiológicos no son impuestos desde el exterior; son, por las características estructuro-determinados de los sistemas, generados internamente.

Más existe un umbral de perturbaciones más allá del cual el sistema se desnaturaliza. Esto ocurre frente a las perturbaciones destructivas.

Por ejemplo, en el control de la temperatura sanguínea, es posible mudar rápidamente de una cámara refrigerada a la plataforma de una fundición de acero, ya que sutiles mecanismos en el organismo ajustarán su temperatura en todo momento alrededor de 37°. ¿Porqué 37°? Ciertamente, no porque alguna "autoridad" haya determinado que así sea, sino porque el organismo mismo reconoce cuál es su propia temperatura óptima de funcionamiento. Del mismo modo, el límite fisiológico es el que el sistema lo reconoce como tal, porque sobrepasarlo sería peligroso.

Asociada a la idea de ultraestabilidad, W.R. Ashby considera la adaptación como una propiedad de la ultraestabilidad. De acuerdo a su definición, la forma del comportamiento es adaptativa si se mantiene las variables dentro de los límites fisiológicos.

Ashby, al hablar de aquellas variables que es preciso mantener dentro de los límites fisiológicos, habla de variables esenciales. Para él, las variables esenciales son aquellas que se encuentran íntimamente ligadas entre sí, tanto que si se produce una modificación considerable de una de ellas, tarde o temprano, se puede observar la modificación de la demás. Por ejemplo, si la frecuencia del pulso de un animal desciende a cero, se puede predecir que la temperatura de su cuerpo se nivelará inmediatamente con la del medio, y que el número de bacterias en los tejidos crecerá de cero a una magnitud mucho más elevada.

La adaptación corresponde a una de las formas de los sistemas de autorregulación. Puede decirse que la autorregulación, la autoadaptación son propiedades fundamentales de todos los sistemas viables, desde la célula hasta los organismos sociales.

5.14 ENTROPÍA.

Se ha mencionado que los sistemas viables son capaces de adaptarse y sobrevivir. Podría agregarse que importan del medio lo necesario para sobrevivir y exportan una cantidad de energía un poco menor que la importada porque, en el proceso de transformación desde entrada a salida, parte de la energía se ocupó en la organización del sistema, en mantener unidas sus partes. Esto lo mide la *entropía*, que en Física es una medida de desorganización correspondiente a la incesante pérdida de energía al interior del sistema.

De acuerdo con este concepto, los sistemas cerrados, que no interactúan con el medio, poseen una pérdida constante de energía, que los lleva a su destrucción o, más específicamente, de vuelta al estado natural de los componentes del sistema. Un ladrillo y una pirámide poseen un destino similar: el primero se transformará en arcilla y la segunda en piedras, sólo es cuestión de tiempo, esto en el corto plazo, tal vez unos cuantos miles de años, porque en el largo plazo debería darse la transformación de los elementos, un átomo cambiará sus propiedades cuando le falte energía para mantener su *orden*.

En contraposición al concepto de entropía está el de *negaentropía*, o entropía negativa, el cual postula que algunos sistemas viables son cada día más organizados y capaces de sobrevivir, producto del uso de energía importada del medio para asegurar la unión de los elementos. Considérese el caso de una empresa que comenzó sus operaciones hace pocos años y ha tenido éxito. Lógicamente, los ejecutivos tenderán a invertir para mantener lo conseguido y para producir más eficientemente, adquirir maquinarias modernas, computadores, contratar especialistas en diferentes áreas, implementar mecanismos de control, etc., lo cual es una política habitual de ir consolidando posiciones, vía organizarse cada vez mejor, para crear bases sólidas desde donde seguir proyectando el crecimiento.

5.15 LA COMPLEJIDAD EN LOS SISTEMAS.

Todo sistema al ser distinguido presenta una organización que le confiere su identidad de clase y una estructura que realiza aquella organización. La estructura queda constituida por los componentes y las correspondientes relaciones entre ellos.

En estas estructuras la complejidad aparece como un rasgo característico de los sistemas. Determinarla implica focalizar la atención fundamentalmente en dos factores: la diversidad de los elementos que conforman el sistema y las interacciones entre aquellos.

En general, desde la perspectiva de un observador, un sistema complejo está constituido por un gran número de componentes o elementos diferentes, dotados de funciones específicas y especializadas. Estos elementos se relacionan según una multiplicidad de interacciones y están ordenados en estructuras estratificadas, tal como ocurre en una perspectiva sistémica vertical con los átomos, las moléculas, las macromoléculas, y las células.

Generalmente las interacciones entre los elementos de un sistema complejo son no lineales, lo cual significa que las variables se relacionan mediante funciones distintas a las de la forma $\alpha x + \beta$, o incluso por medio de funciones de otras variables.

De acuerdo a estas consideraciones, la complejidad aparece asociada a la diversidad de los elementos y a la no linealidad de las interacciones, en el marco de una totalidad organizada.

De todo ello resulta un comportamiento muy particular de los sistemas complejos, que se caracteriza fundamentalmente por su irreductibilidad, por la emergencia de nuevas propiedades en cada uno de los sucesivos niveles sistémicos de la estructura estratificada. Además, es esta complejidad la que le confiere la platicidad y adaptación al medio.

La complejidad aparece entonces como una característica fundamental de los sistemas. Se traduce principalmente en la incapacidad de describir el sistema como un todo, y de deducir su comportamiento, a partir del conocimiento de la conducta de sus componentes.

5.16 MEDICION DE LA COMPLEJIDAD Y CONCEPTO DE VARIEDAD.

Un sistema aparece cuando una unidad simple distinguida en un comienzo, mediante una distinción adicional, es descompuesta por el observador en componentes tales que al componerse generan la unidad simple original. Aparece así un conjunto de elementos en interacción.

En la mayoría de los sistemas las interacciones entre componentes son no triviales, más aún, son ellas las que confieren al sistema su carácter complejo.

En este contexto, y entendida la complejidad de un sistema como fue explicada, su determinación obviamente no aparece como un asunto trivial; sin embargo, considerar como medida de la complejidad el número de estados posibles del sistema aparece como un acercamiento razonable, más aún cuando, conceptualmente, aquel criterio de medida corresponde a la noción de variedad, empleada por Asbhy, y por Stafford Beer, entre otros, al hacer referencia al problema de la complejidad de los sistemas.

Stafford Beer, refiriéndose a la complejidad, señala que "... si esta magnitud ha de describirse razonablemente, hay que medirla. Se necesita un concepto de complejidad claramente formulado y su nombre científico es variedad".

Cabe destacar que todo intento de explicitar el estado de un sistema necesariamente debe prestar atención privilegiada a las relaciones que manifiestan sus elementos, más aún cuando estos, a su vez, están determinados por la interacción de las componentes que lo conforman, razón por la cual también adquieren la categoría de sistemas, aunque respecto del todo son, en esencia, sub-sistemas.

Sin embargo, la variedad será entendida como referida a un sistema que se expresa a un nivel sistémico.

En una primera aproximación, la variedad puede concebirse como una magnitud proporcional al número de elementos distinguibles del sistema, o de otra forma,

variedad := μ (complejidad del sistema), con μ = criterio de medida, y

variedad \propto número de elementos distinguibles del sistema.

Por lo tanto, una manera de proceder para poder determinar, en alguna medida, la variedad de un sistema, es identificar los elementos distinguibles y contar esos elementos. Pero obviamente, esta forma de medir el número de estados posibles del sistema y de este modo su complejidad, es parcial e inexacta, ya que se está obviando un factor determinante en la complejidad, como lo es la interacción entre los elementos.

Para esto es fundamental conocer la profundidad con que el observador "penetra" en la naturaleza del sistema que distingue, tras haberlo definido en la distinción.

Al intentar determinar la variedad de un sistema, puede optarse por reducirlo como un fenómeno, o bien considerar el sistema en su verdadera dimensión, es decir como un conjunto de elementos en interacción. Esto implica enfrentar dos alternativas: reducir el sistema a una colección de elementos, o bien, considerarlo como una unidad compuesta, esto es, como una asociación sinérgica de elementos. La diferencia entre ambas opciones obviamente radica en el reconocimiento de las relaciones existente entre los elementos.

En este sentido, es importante destacar el rol del observador en la determinación de la variedad. Ésta no es una propiedad intrínseca del sistema, sino que está sujeta a la definición del observador y a su poder de discriminación.

Por lo tanto, y en una perspectiva más amplia, se puede afirmar que la variedad de un sistema es proporcional al número de estados posibles y distinguibles que el sistema puede alcanzar, en donde hablar de estados posibles y distinguibles implica necesariamente enfrentar diferentes formas de determinar la variedad.

Estas diferentes formas se agrupan en cinco casos.

- **Variedad de un conjunto de elementos diferentes.**

Para un conjunto de elementos diferentes, como aquel se presenta en la Fig. 6, la variedad es $VAR = 7$. En este caso, existen siete elementos identificables.

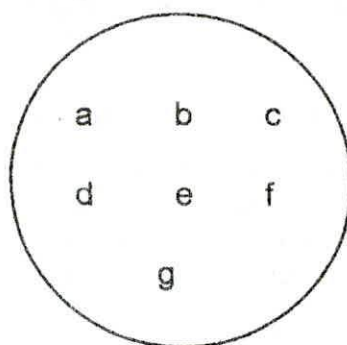


FIGURA 6.

Cabe destacar que en el ejemplo de la Fig. 6, para determinar la variedad basta con la información que ella provee.

Por lo tanto, en este caso, la variedad es:

$$VAR = n$$

donde, n = número de elementos.

- **Variedad de un conjunto de elementos parcialmente diferentes.**

En un conjunto de elementos parcialmente diferentes, como aquel que se presenta en la Fig. 7, la variedad es:

$$\text{VAR} = m$$

donde m = número de elementos diferentes.

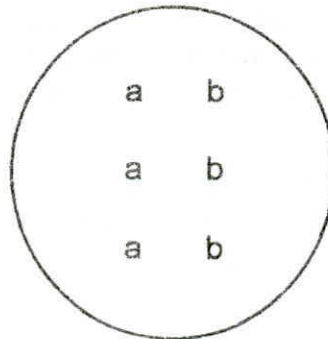


FIGURA 7.

En este caso la variedad es $\text{VAR} = 2$. Ello es así porque el universo real que aparece en la Fig. 7 sólo tiene dos elementos diferentes: a y b , los otros elementos o son a o son b .

Se puede observar que en este caso se requiere de información adicional para determinar la variedad. Esta información se refiere a la semejanza entre los elementos.

- **Variedad de un conjunto de elementos diferentes relacionados.**

En un conjunto de elementos diferentes relacionados la atención ya no aparece ligada a los elementos. Es necesario ahora observar las relaciones entre ellos, ya que se trata de un conjunto en el cual la ocasión aparece como el factor fundamental y por lo tanto determinante en la pretensión de establecer la variedad.

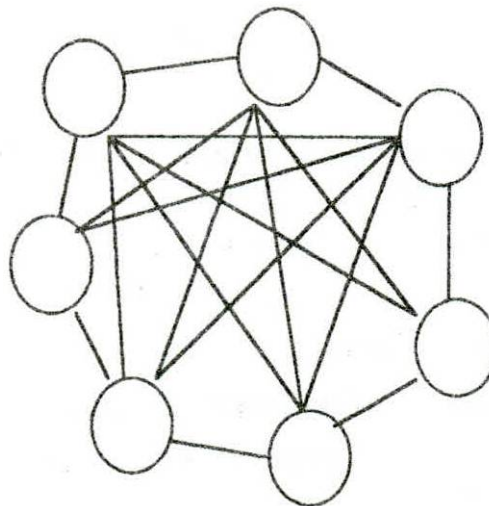


FIGURA 8.

En este caso, si se tiene un sistema en el cual sólo se sabe que cada elemento se relaciona con los otros restantes, la relación en un sentido será la misma que la relación en el otro sentido. Esto es, la relación $B-F = F-B$, como ilustra la Fig. 8. En ella, cada punto se liga con los otros seis, de modo que existen 42 relaciones. Pero cuando se especifica una relación, por ejemplo, entre B y F, se está indicando que toda relación entre aquellos dos puntos ya está establecida. Por lo tanto, en este caso, si se desea especificar la relación entre F y B, obviamente ella ya está determinada. En términos cuantitativos esto significa que la mitad de las 42 relaciones establecidas no tiene que considerarse al determinar la variedad. En este caso, entonces, la variedad, esto es, el número de modos en que se pueden relacionar "n" elementos es:

$$\text{VAR} = n(n-1) / 2$$

donde n = número de elementos relacionados.

Por lo tanto, en el caso de la Fig. 8, la variedad es $\text{VAR} = 21$.

En este caso, se requiere de mayor información adicional. Se requiere conocer el número de elementos, y el tipo de relación de cada elemento; en el caso de la Fig. 8, ésta es el tipo "de uno a todos".

- **Variedad de un conjunto de elementos diferentes relacionados sistemáticamente.**

En este caso, mas allá de la cohesión entre los elementos, existe un tipo especial de relaciones que está dado por la orientación de las relaciones entre aquellos elementos.

Es claro que este caso es similar al anterior, sólo que ahora el número de relaciones crece y también la variedad, ya que una relación, por ejemplo, entre F y B, no es la misma que entre B y F. Por lo tanto, operacionalmente, para determinar la variedad no es preciso dividir por dos.

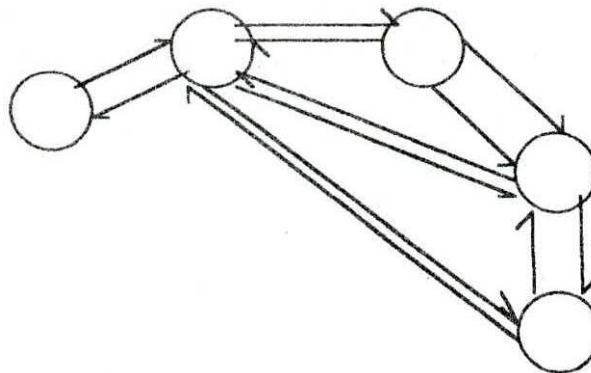


FIGURA 9.

Esta situación aparece en la Fig. 9. En ella la variedad corresponde a:

$$\text{VAR} = n(n-1)$$

donde n = número de elementos relacionados.

Lo que ocurre en este caso es que al disponer de mayor información, la variedad aumenta. Ahora la relación B-F es diferente de la relación F-B. En el caso anterior,

sólo se conoce que entre los puntos F y B existe una relación, pero nada respecto de la relación en sí, como ocurre en esta situación.

Una relación característica de este tipo es la relación de parentesco. Por ejemplo, si Juan es hijo de Flavio y Flavio es hermano de José, entonces José es tío de Juan.

Sin embargo, también Juan es sobrino de José. En el caso anterior la relación existente sólo es de parentesco.

Esta es una relación general que destruye la mitad de la información de que se dispone para reconocer la relación tío-sobrino.

Las relaciones en un conjunto de elementos distintos asociados sistemáticamente pueden adoptar muchos valores y no sólo dos. En el ejemplo, José puede ser mas alto que Juan, y Juan mas bajo que José. También José puede ser mas viejo que Juan, y Juan mas joven que José, y así, es posible encontrar diferentes relaciones casi indefinidamente.

Por cada nueva clase de información que se agrega, el número de relaciones posibles se eleva y con eso la variedad aumenta. En este caso, para determinar lo que realmente ocurre, es preciso eliminar una mayor cantidad de incertezas.

- **La variedad de un sistema dinámico.**

Se puede decir que este es el caso que corresponde más exactamente a la noción de sistemas. Esto quiere decir que los siete elementos considerados en los casos anteriores, no solo manifiestan cohesión, además, sus relaciones expresan un padrón. Inclusive, desde la mirada de un observador puede decirse que están unidos por un objetivo común. En este sentido, los casos anteriores corresponden más bien a conjuntos estáticos de elementos.

Los sistemas viables en general pueden considerarse sistemas dinámicos, es decir expresan comportamientos caracterizados por mantención de organización en continuo cambio estructural. Entre las consecuencias que se derivan de este comportamiento se puede mencionar el cambio de estado de las relaciones que definen el sistema. De este modo, si una cierta relación en un sistema no puede adoptar más que un estado, entonces el sistema como tal no puede cambiar de estado. Esto puede significar imposibilidad de hacer, por lo cual la existencia de actos determinando sus conductas se vuelve no posible. Por ejemplo, en un automóvil, las conexiones eléctricas del motor de combustión interna tienen que hacerse efectivas para comenzar a funcionar.

En el sistema de la Fig. 10, para determinar la variedad, se necesita conocer primero el número de elementos diferenciables. Pero, ya que se está considerando la dinamicidad como característica fundamental, los elementos corresponderán a los estados del sistema. Puede ocurrir que para un cierto estado la relación, por ejemplo, A-G esté activa entanto que las restantes relaciones se encuentran no activadas. Un segundo estado puede corresponder a la situación anterior, pero con la relación A-G desconectada. Un tercer estado puede tener ligada la relación G-A y un cuarto estado puede tenerla desligada. Puede apreciarse que estos cuatro estados se derivan de las posibilidades de relación existente entre dos elementos.

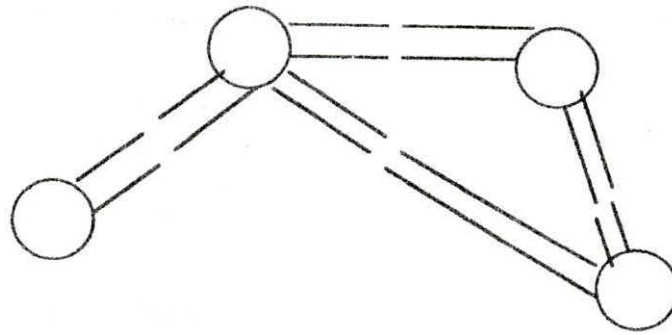


FIGURA 10.

Es el caso ejemplificado, es constatable que existen dos relaciones y dos opciones para cada una de ellas. Es decir, $2 \cdot 2 = 4$. Ahora, si se introduce otra conexión, habrán tres relaciones y dos opciones para cada una de ellas, esto es, $2 \cdot 2 \cdot 2 = 8$.

Como existe 42 relaciones, el número de estados distinguibles corresponderán a 42 veces 2, repetidos como factores, o sea, 2^{42} .

Por lo tanto, la variedad en este caso corresponde a:

$$\text{VAR} = 2^{(n(n-1))}$$

En términos generales, puede decirse que la variedad de un sistema es proporcional al número de estados posibles y distinguibles que el sistema puede alcanzar.

La variedad, como lo expresa S.Beer, es una unidad de medida que puede considerarse como una aproximación razonable para tener alguna noción acerca de la complejidad de un sistema. Puesto que se está tratando con una medida, y toda medida debe expresarse en un denominador común, la variedad adopta una unidad estandarizada, a fin de permitir la posibilidad de comparar las variedades de distintos sistemas. Esta unidad es el "bit" (binary digit en inglés), que en realidad es adimensional. Para poder expresar la variedad operacionalmente es necesario aplicar logaritmo en base dos al número de estados posibles del sistema, es decir,

$$\text{variedad (sistema)} := \log_2 (\text{número de estados posibles del sistema}) [\text{bits}]$$

5.17 CLASIFICACION DE LOS SISTEMAS

Los sistemas han sido clasificados por múltiples investigadores y desde diferentes puntos de vista. Aparece así una vasta lista de clasificaciones, las cuales, en muchos casos, más que constituirse en un aporte clarificador, llevan más bien a confusiones. En este contexto, entre las clasificaciones más relevantes, es posible distinguir aquellas que se basan en la complejidad, en su determinismo, en su relación con el medio, en su origen, entre otras.

Entre todas las propuestas se presentarán aquellas que han tenido una mayor difusión en la literatura que aborda los problemas referidos a sistemas.

1. En 1956 K. Boulding presentó un catálogo, que puede ser calificado más bien de informal, en relación a los niveles principales en la jerarquía de los sistemas. Su proposición explicita los siguientes niveles sistémicos:
 - 1.1. Sistemas a nivel de estructuras estáticas o marco de referencia. Corresponden a un primer nivel de complejidad, como por ejemplo lo es la anatomía y geografía del universo, un edificio, un átomo, una molécula, los cristales y las estructuras biológicas desde el nivel microscópico electrónico hasta el macroscópico.
 - 1.2. Sistemas dinámicos simples. Son aquellos sistemas que poseen movimientos fijos y predeterminados, como por ejemplo: un motor eléctrico, el sistema solar, un reloj, la máquina ordinaria en general, etc.
 - 1.3. Sistemas con mecanismos de control o sistemas cibernéticos. Son aquellos sistemas que se caracterizan por la transmisión e interpretación de información. En ellos existe un elemento controlador del sistema, que mantiene un determinado estado basado en la información sobre el estado interno, y un cierto objetivo esperado, de tal modo que establece las medidas correctivas necesarias toda vez que el estado interno se aleje del objetivo deseado.

Como ejemplos se pueden señalar, los termostatos, los controladores de velocidad, los servomecanismos, los mecanismos homeostáticos en la organización, entre otros.
 - 1.4. Sistemas abiertos autosostenidos. Son aquellos sistemas que se caracterizan por la automantenimiento de la estructura y la capacidad de autorreproducirse.

Una característica relevante de estos sistemas es que en ellos la vida comienza a diferenciarse de la no vida.

Como ejemplo de este tipo de sistema se puede mencionar, la célula, y los organismos vivos, en general.
 - 1.5. Sistemas Genético-sociales. Son aquellos que corresponden al nivel de las plantas u organismos "vegetaloides". Se caracterizan por la división de labores entre las entidades simples que lo conforman, estableciéndose cierta formación de "sociedades", como lo son por ejemplo las asociaciones de células a nivel de raíces, hojas, semillas, etc. en un árbol.

Además, estos sistemas se caracterizan por su equifinalidad, es decir, los subsistemas que lo conforman se orientan a un mismo objetivo cualesquiera sean sus estados iniciales. También poseen receptores de información difusos y limitados. Esto ocurre, por ejemplo, con el girasol.
 - 1.6. Sistemas a nivel animal: Los sistemas que pertenecen a esta categoría se caracterizan por un incremento de la movilidad en comparación con el nivel anterior, la aparición de la conciencia se asume más bien a nivel de "instinto", la capacidad de interpretar símbolos simples, como gritos de advertencia, y otros, y por su conducta teleológica determinada por objetivos.

Además, en este nivel se produce una especialización de los receptores de información, como los ojos, oídos, y otros, lo que permite la existencia de un sistema de información, el sistema nervioso que facilita la toma de ciertas decisiones específicas.

- 1.7. Sistemas a nivel humano: En este tipo de sistemas se pueden observar, además de las características correspondientes al nivel animal, la aparición de una conciencia altamente elaborada, la producción e interpretación de símbolos complejos y la elaboración de imágenes de tiempo y relación, siendo éstas, fundamentalmente complejas y reflexivas.
- 1.8. Sistemas a nivel de organizaciones sociales: Estos sistemas se caracterizan por su complejidad, la cual está dada por la interacción y organización de elementos pertenecientes al nivel de los sistemas humanos con sus correspondientes características y consecuente complejidad.

La naturaleza de los elementos de tales sistemas hace destacar la importancia de la coordinación, comunicación, y unidad de dirección de tales elementos. En este nivel es posible distinguir todas las comunidades existentes.

- 1.9. Sistemas trascendentales: Son los sistemas superiores absolutos, inevitables, pero ignorados, o conocidos sólo parcialmente.

Ludwig von Bertalanffy señala que este repaso es impresionista e intuitivo y no aspira al rigor lógico. Por reglas general, los niveles superiores presuponen los inferiores (por ej. los fenómenos socio-culturales, el nivel de la actividad humana, etc.) pero la relación entre niveles requiere aclaración en cada caso (problemas como el del sistema abierto y el código genético como aparentes requisitos previos para la "vida", la relación entre sistemas "conceptuales y redes", etc.). En este sentido, la lista insinúa tanto los límites del reduccionismo, como los vacíos en el conocimiento actual.

2. Otro criterio de clasificación se basa en la posibilidad de predecir su comportamiento, de acuerdo al cual los sistemas se pueden clasificar en:

- 2.1. Sistemas determinísticos: Son aquellos sistemas cuyo comportamiento es dinámico, pero totalmente predecible, es decir es aquel en el cual las partes interactúan en forma perfectamente predecible, como ocurre, por ejemplo, con un electrón.
- 2.2. Sistemas probabilísticos: Son aquellos cuyo comportamiento no se puede predecir. Como ejemplo, se puede citar a un operador más su computador.
Es posible, también, considerando en conjunto los criterios de clasificación basados en la posibilidad de predecir su comportamiento y en la complejidad que manifiestan, establecer una nueva diferenciación en los siguientes términos.
- 2.3. Sistemas complejos determinísticos: Son aquellos sistemas sumamente complejos (alta variedad de elementos e interacciones entre tales elementos),

de comportamiento dinámico, y totalmente predecible, como por ejemplo, el computador.

- 2.4. **Sistemas simples probabilísticos:** Son sistemas simples (pocos componentes e interacciones entre ellos), pero de comportamiento no predecible, como por ejemplo, el lanzamiento de una moneda.
- 2.5. **Sistemas complejos probabilísticos:** Son sistemas complejos, es decir, altamente elaborados e interconectados, cuyo comportamiento no puede ser predecido, y por lo tanto se pueden caracterizar en términos probabilísticos. Un sistema de mantención de inventario, el reflejo condicionado de un animal, son ejemplos de este tipo de sistemas.
- 2.6. **Sistemas excesivamente complejos y determinísticos:** Son sistemas excesivamente complejos, con muchos componentes, no todos conocidos, pero con comportamiento previsible. Es posible afirmar que en la realidad no existen sistemas de este tipo.
- 2.7. **Sistemas excesivamente complejos y probabilísticos:** Son aquellos sistemas excesivamente complejos, con muchos componentes, no todos conocidos, cuyo comportamiento no puede predecirse en forma exacta.

3. Un tercer criterio de clasificación se basa en clase a la que pertenecen las cantidades observadas o medidas del sistema. De acuerdo a este criterio existen:

- 3.1. **Sistemas físicos:** Son aquellos cuyas cantidades son medibles. Así mismo, dentro de estos sistemas se pueden distinguir:
 - 3.1.1. **Sistemas físicos reales:** Aquellos en los cuales las cantidades involucradas pertenecen a la realidad.
 - 3.1.2. **Sistemas físicos conceptuales:** Aquellos en los cuales las cantidades involucradas son asumidas.
- 3.2. **Sistemas abstractos:** Son aquellos cuyos elementos son conceptos, por lo tanto, no medibles. En este tipo de sistemas, los elementos son creados por definición, y las relaciones por proposición, es decir, se conforman por axiomas y postulados.

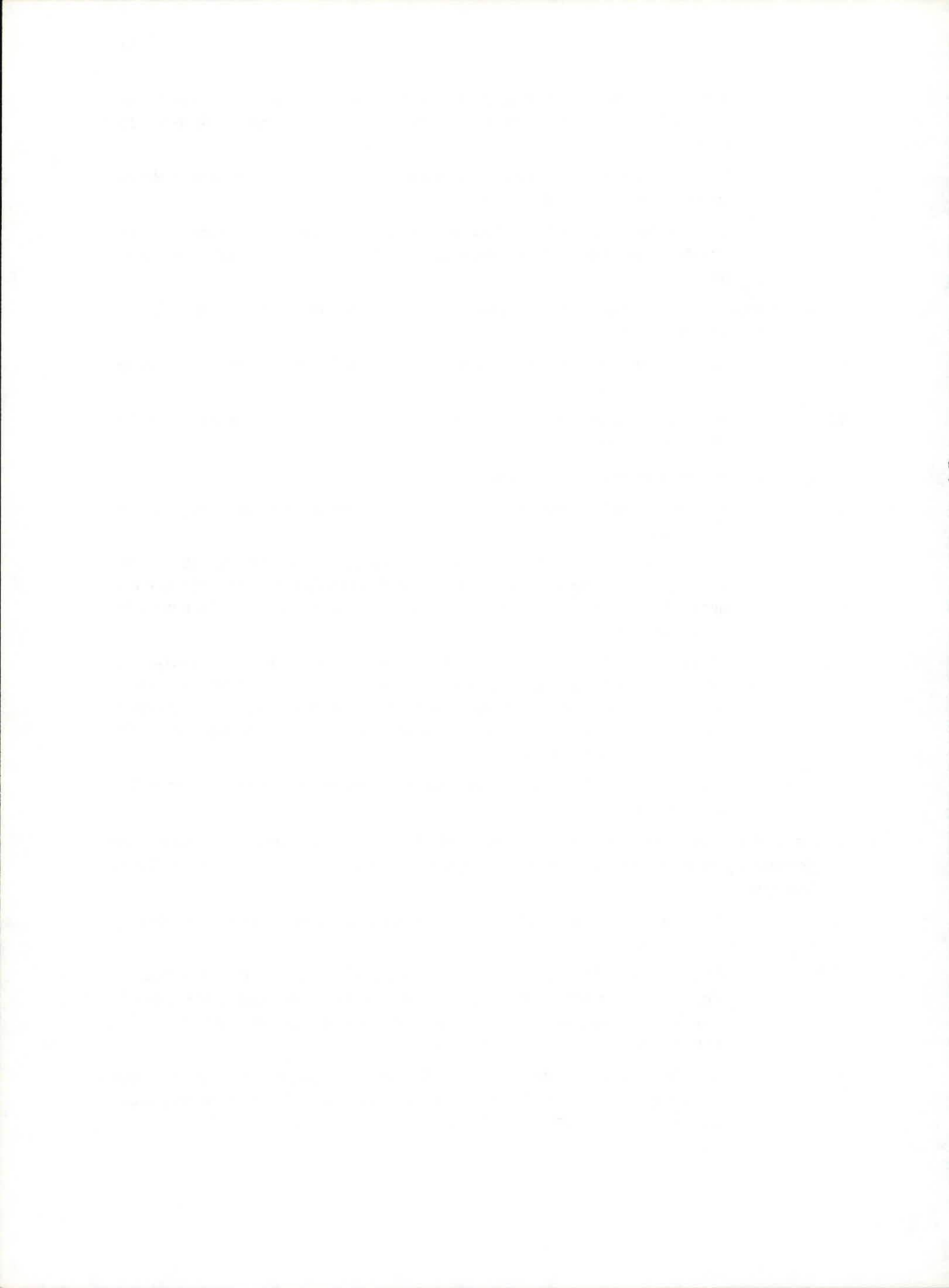
Ejemplos de este tipo de sistemas, son algunos sistemas que pertenecen al dominio de la Matemática, como grupos, espacios vectoriales, etc.
- 3.3. **Sistemas limitados:** Son aquellos que poseen un número finito de cantidades externas y una estructura finita. A esta categoría pertenecen los sistemas físicos reales.
- 3.4. **Sistemas ilimitados:** son aquellos en los que, tanto las cantidades externas a medir como su estructura son infinitas.

Cabe señalar que los sistemas físicos conceptuales y abstractos, pueden ser considerados como limitados o ilimitados.

- 3.5. Sistemas discretos: Son aquellos en que las variables pueden tomar un número finito de valores discretos, y son conocidos en instantes discretos de tiempo.
 - 3.6. Sistemas continuos: Son aquellos en los que las variables adquieren valores continuos, en un tiempo continuo.
 - 3.7. Sistemas híbridos: Son aquellos en los cuales las variables adquieren valores continuos, pero sus valores son conocidos solamente en instantes discretos de tiempo.
4. Un cuarto criterio de clasificación es el basado en su origen. De acuerdo a este criterio, los sistemas se pueden clasificar en:
- 4.1. Sistemas artificiales: Son aquellos elaborados por el hombre. Como ejemplo, un computador.
 - 4.2. Sistemas naturales: Son aquellos que surgen como tales sin la participación explícita del hombre.
5. Según intercambio de energía con el medio.
- 5.1. Sistemas cerrados: Son aquellos que no mantienen intercambio alguno con su medio.

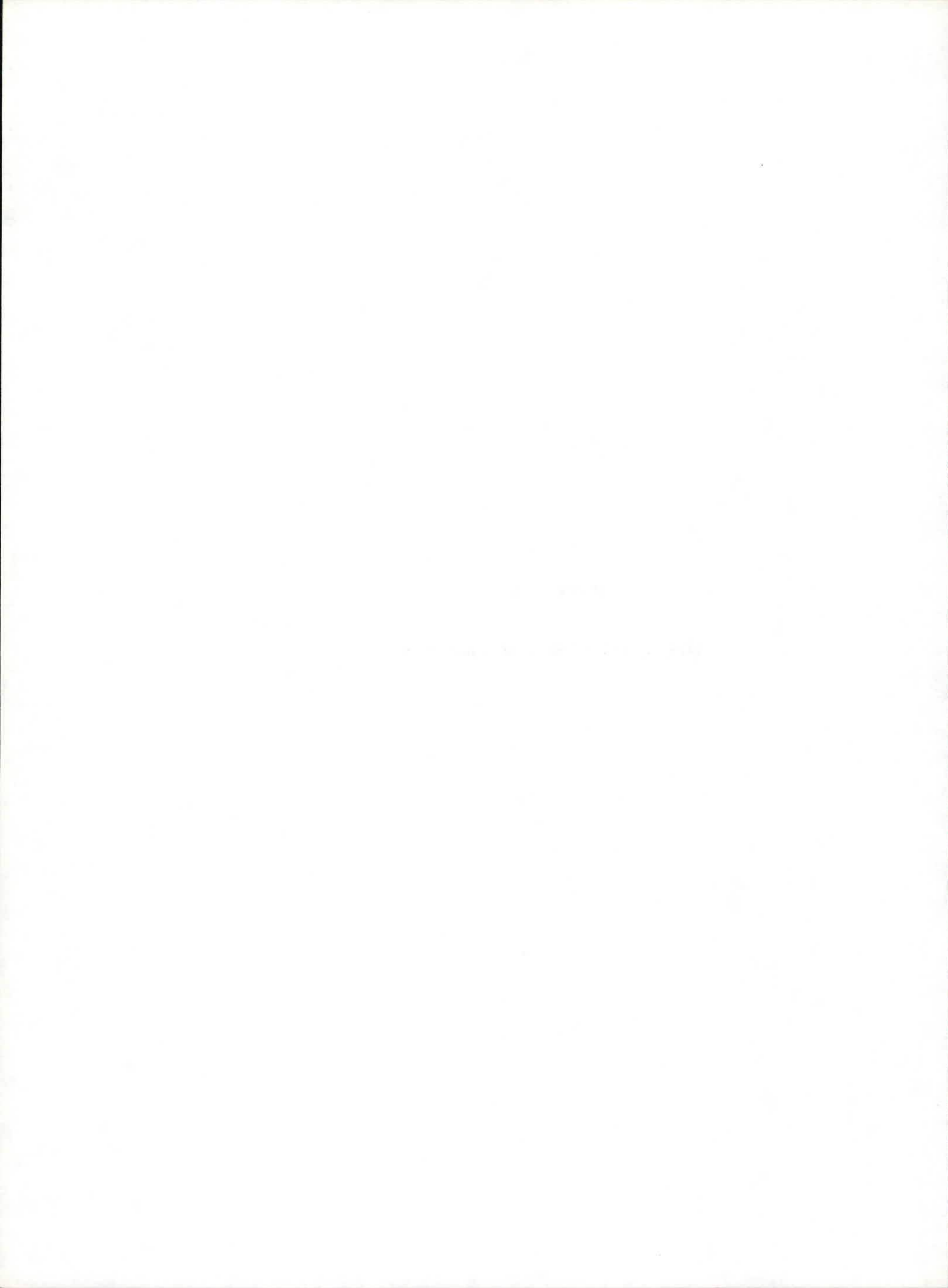
En tales sistemas, vistos desde la perspectiva de sus intercambios de materia y energía con el medio, es decir, no desde el punto de la organización que los define, la entropía (tendencia al caos) aumenta, llevando inevitablemente a la destrucción del sistema.
 - 5.2. Sistemas abiertos: Son aquellos que intercambian materia, energía e información con su medio ambiente. Debido a este intercambio, estos sistemas son capaces de contrarrestar el aumento de entropía, pudiendo mantenerse, e incluso evolucionar, hacia estados de menos entropía, o de mayor orden y complejidad.

Como por ejemplo de este tipo de sistemas se puede mencionar a los organismos vivos.
6. Clasificación de sistemas según su complejidad. Este criterio se basa en el número de elementos y en el número de interacciones que manifiesta un sistema. En base a ello se distinguen:
- 6.1. Sistemas dinámicos simples: Son aquellos que tienen pocos componentes y pocas relaciones.
 - 6.2. Sistemas complejos descriptivos: Son aquellos altamente elaborados, es decir, conformados por muchos componentes muy interconectados, pero de relaciones conocidas y por ende descriptibles como sistemas. Un computador es un ejemplo de un sistema de este tipo.
 - 6.3. Sistemas excesivamente complejos: Son aquellos que, por no conocer todos sus componentes, y por lo tanto tampoco todas sus relaciones, no se pueden describir en forma precisa y detallada.



SECCION B

LOS SISTEMAS ORGANIZACIONALES



ALGUNOS ASPECTOS DE LA TEORIA DE LA REGULACION.

La homeostasis es un tipo de autocontrol que opera en los sistemas ultraestables, donde la existencia de circuitos de retroalimentación permite que ese autocontrol sea alcanzado.

Algunos autores al hablar de retroalimentación, hablan de servomecanismo, como Paul Idatte, (IDATTE-1972). No obstante el término servomecanismo se emplea preferentemente en relación a la idea de "mecanismo de servicio", esto es, un instrumento que sirve para controlar determinado proceso técnico, por ejemplo, una máquina, un conjunto de máquinas, instalaciones eléctricas, etc. En otras palabras el término se utiliza para definir el mecanismo que al substituir al hombre actúa sobre determinada máquina o proceso técnico, (LANGE-1972).

Es la aplicación cada vez mayor de estos servomecanismos a los procesos técnicos lo que genera la necesidad del análisis matemático -con un significativo avance ya en la década de los 40- de su funcionamiento y, precisamente, a este análisis se le denomina teoría de la regulación, teoría general de la regulación, teoría del control automático, e inclusive, teoría de los servomecanismos, y que corresponde a la teoría matemática en que se apoya la automatización de los procesos tecnológicos.

No obstante, lo que importa es que la retroalimentación es el concepto básico de esta teoría del control, teoría cuyo objeto de estudio es precisamente el control y que será denominada "Teoría general de la regulación".

Hoy se verifica, como dice (MASER-1975), la transición de la técnica de la regulación para una Teoría general de la regulación y para las aplicaciones de esa teoría a procesos biológicos, sociales, económicos, de comunicación, etc.

Por lo tanto, y pensando en las Organizaciones de actividad humana, la Teoría general de la regulación, puede constituir un aporte para proyectar organizaciones homeostáticas.

Interesa, por lo tanto, revisar los aspectos mas relevantes de aquella teoría.

El control, como función, aparece cuando se requiere que una propiedad o variable, que caracteriza una operación o un proceso, asuma un determinado valor o un conjunto de valores.

Estos, por su vez dependen de los valores que asuma una otra variable, siempre que exista, entre ellas, una determinada relación de dependencia susceptible de ser representada por un modelo. Al variar una de ellas se puede controlar los valores de la otra. En la figura 1 se representa esta situación, usando la nomenclatura utilizada: la variable primera, que se desea

pueda asumir valores determinados, recibe el nombre de "variable de salida". La otra es la entrada o señal de control.

Cuando se conoce la relación entrada/salida y esta no muda, es suficiente con que la entrada asuma los valores correspondientes. Esto se consigue directamente o por un controlador apropiado. Esta forma de control se conoce como de lazo abierto. Cabe destacar que algunos autores la denominan sistema de control o simplemente "Control", en cuanto sistema de control.

Esta forma de control -sistema de control- presenta muchos problemas ya que al elaborar las ordenes, no se conoce si el proceso alcanzará en su salida los valores deseados. En este caso, no se aplican los principios de la autorregulación. Es posible que el comportamiento del proceso se represente a través de una expresión matemática que se conoce como modelo, que no es una representación exacta, sino simplificada del comportamiento real, válido sólo si el proceso funciona en ciertas condiciones. Esto es, que no ocurran factores que afecten el comportamiento del proceso. Estos factores se denominan "perturbaciones". Mas, aquel factor que representa el trabajo a realizar por la operación o proceso y que puede sufrir variaciones (volumen de líquido contenido en un calentador, por ejemplo), se denomina carga de proceso. Todo eso puede hacer que el valor real de la salida no sea el deseado. La figura 2 ilustra esta situación.

La forma de conseguir que la salida no sea afectada por las perturbaciones, por las variaciones de carga, inexactitudes o simplificaciones en el modelado, permitiendo inclusive la corrección de desvios, es utilizar la salida para generar las ordenes o valores de entrada, para lograr de este modo que el sistema, además de corregir los desvios producidos, alcance los objetivos que pretende. Se tiene, así, una forma de control -sistema de control- en circuito cerrado, sistema de control denominado por algunos autores simplemente, Regulación.

Sus elementos aparecen en la figura 3. La salida del proceso se mide a través de un dispositivo adecuado comparando su valor con el valor deseado, denominado "referencia". La diferencia entre los dos valores es el error, que, por su vez, es la señal de entrada a un dispositivo que calcula el valor que debe tener la entrada o control para cada error. Este dispositivo recibe el nombre de "controlador" o "regulador". La señal que genera, es enviada al "actuador", que es el encargado de producir aquella entrada.

Operacionalmente, se puede representar estas operaciones en un esquema como aquel de la figura 4.

Los elementos, en este sistema de control, tiene las siguientes funciones:

- referencia (R): corresponde a una señal externa al sistema de control, cuya función es el comando de un efecto específico en el

sistema de control, y que en la mayoría de los casos se refiere a la la condición "ideal" de funcionamiento del sistema. Esto es, representa el valor numérico que se pretende alcanzar con la variable controlada (C).

- punto de mezcla: es el lugar en que se realiza la "suma algebraica" de señales: el valor de referencia (R) y la señal de retroalimentación (+/- b), que puede ser de signo positivo o negativo.

- señal de error (e): es el resultado de la suma algebraica realizada a nivel de punto de mezcla.

- sistema de control (G1) o controlador: es el que genera la señal (m) para "controlar" el sistema controlado (G2).

- variable manipulada (m): corresponde a la cantidad o condición que el sistema de control (G1) aplica al sistema controlado (G2);

- sistema controlado (G2): es el proceso sujeto a control.

- variable de salida o controlada (C): es la variable del sistema controlado que está sujeta a control.

- elemento de retroalimentación (H): es el componente que establece la relación funcional entre la variable de salida (C) y la señal primaria de retroalimentación (+/- b).

- señal primaria de retroalimentación (b): es una función dependiente de la señal de salida (C), que previamente ha sido modificada por el elemento de retroalimentación (H), de modo que $b = CH$. La señal (+/- b) se suma algebraicamente, en el punto detector de error, al valor de referencia (R), para obtener la señal de error (e).

- perturbación (+/- u): es la señal de entrada de naturaleza aleatoria, generalmente indeseable, que afecta el funcionamiento del proceso a controlar (G2) y se refleja en modificaciones de la variable de salida (C). La perturbación (+/- u) puede ser positiva o negativa y sumarse algebraicamente con la señal (m); en algunos casos existe un segundo punto detector de error en el trayecto donde estas variables se suman algebraicamente (m +/- u).

- vía ortodrómica: es la línea de relaciones que comienzan con el valor de referencia (R) y termina con la variable controlada (C).

- vía antidrómica o de retroalimentación: es la transmisión de la señal desde la variable controlada (C) hasta la señal primaria de retroalimentación (+/-), previa transformación a nivel del elemento de retroalimentación (CH).

En resumen, el Control en circuito abierto -sistema de control- se constituye en un modelo capaz de producir efectos, de

forma abierta, permitiendo así hablar con mayor precisión, de circuito de control. La Regulación -sistema de control- es un modelo para producir efectos, pero de forma cerrada, permitiendo que se hable, con mayor precisión, de circuitos de regulación. El factor esencial en cada circuito de regulación es el acoplamiento retroactivo, o retroalimentación.

La homeostasis opera la través de sistemas de control en circuito cerrado, específicamente a través de circuitos de retroalimentación. Sin embargo, la teoría de la regulación no solo contempla estos sistemas de control. Se refiere también a los los sistemas de control de circuito abierto. Por lo tanto los dos sistemas serán considerados.

SISTEMA DE CONTROL EN CIRCUITO CERRADO.

En un circuito de regulación, el sistema total se autocorriga de los posibles desvíos, a través de la introducción de aquel acoplamiento retroactivo. Esta retroalimentación puede ser negativa ou positiva.

En estos sistemas cuando los datos contribuyem a acelerar los comportamientos en el mismo sentido de las conductas precedentes -el retorno se suma a la referencia- se trata de un lazo de retroalimentação positivo ya que sus efectos son acumulativos.

Se os datos actúan en sentido opuesto al que manifiestan los comportamientos anteriores - el retorno se resta de la referencia- se trata de un lazo de retroacción negativo, ya que sus efectos estabilizan el sistema.

Puede decirse que en un sistema a controlar hay retroalimentación cuando el comportamiento de la transformación es medido y su valor comparado con el valor deseado -la referencia- para establecer la diferencia o error, la cual, por su parte es la señal de entrada de un controlador o regulador que por su vez determina la entrada de control para la transformación.

LA RETROALIMENTACION NEGATIVA.

Un ciclo de retroalimentación negativa es aquel que conduce a un comportamiento adaptativo. Por lo tanto en él existe una convergencia hacia un fin, tal como ocurre en un reservatorio hidráulico, en el cual el nivel de agua se mantiene constante a través de la acción de una esfera reguladora unida al registro de entrada del agua.

Um ejemplo esclarecerá mejor esta situación.

En este caso, si se retira agua del vaso T, se altera, por la salida del líquido, el nivel de agua X, bajando su nivel. Como se procura mantener constante el referido nivel, debe haber entrada de agua, por el conducto W, mediante la abertura de la

válvula Y. Se establece, de esta forma, una relación directa entre el valor X, que se desea regular, y el valor que corresponde a Y, que se fija mecánicamente, con simplicidad, por medio de una boya S, cuya posición permite afectar, a través de una alavanza, la entrada del agua, en la válvula Y, provocando la alteración del nivel X. Dicho de otra manera, bajando el nivel de agua, X, la boya también desciende, lo que provoca la abertura de la válvula Y. Subiendo el nivel, sube también la boya S, lo que provoca el cierre de la válvula Y. Cuanto mayor es la cantidad de agua retirada, tanto más rápidamente y más ampliamente se abre la válvula. Así, el nivel de agua permanece razonablemente igual, con un valor fijo. Está claro que el lugar de la boya puede ser ocupado por una persona, con lo cual se está en lo que se denomina regulación manual. La ventaja de la regulación reside en que perturbaciones no previstas pueden ser consideradas, ya que la abertura de la válvula está bajo influencia directa del nivel de agua. En este caso, se llega a una situación como la que se presenta en la figura 5.

En este caso se trata de un lazo de retroalimentación negativa, ya que existe una tendencia al comportamiento adaptativo o propositivo, esto es, parece tender hacia lo que en este caso es la mantención del nivel de agua en el depósito. Se puede decir que el efecto tiene tendencia a actuar sobre su causa para atenuarla.

Para ello existe un mecanismo autoregulator que oscila indefinidamente alrededor de una posición de equilibrio estable, que además es posible de corregir para reajustar su posición.

El ciclo de retroalimentación puede conformarse por un sistema más complejo que la boya del depósito. A través de este sistema circula información, esto es, la señal que permite desencadenar una acción. La información puede transmitirse a través de un cable eléctrico, por una onda electromagnética, o por un sonido.

En consecuencia, la regulación por retroalimentación actúa siempre a partir de un efecto de salida, hacia un efecto de entrada. En el caso del ciclo de retroalimentación negativa, siempre existe un fin, hacia el cual debe tender el comportamiento del sistema.

En algunos casos, este fin es establecido por el propio sistema y se conserva durante su evolución, como ocurre por ejemplo, en la mantención de la temperatura de la sangre en los organismos vivos. En otros casos es impuesto, como ocurre cuando el hombre los confía a las máquinas por él creadas. En estos casos, cada vez que la conducta real se aleja de la conducta esperada, se genera una acción correctiva para restablecer el equilibrio. Esto es, hay regulación donde el sistema oscila alrededor de una posición de equilibrio tal como aparece en la figura 6.

Joel de Rosnay dice, "en un lazo negativo, toda variación

hacia el, implica una corrección hacia el menos (e inversamente)", ROSNAY (1977).

Las características principales del lazo de retroacción negativa son:

- Tender al equilibrio, a la estabilización.
- Compensar los desvíos.
- Tender hacia la convergencia
- Permitir la readaptación interna.
- Permitir un comportamiento orientado hacia un fin o una meta.

Esta acción autocorrectora de la retroalimentación se encuentra resumida en cuadro 1.

COMPARADOR		CONTROLADOR	
comparación	error	entrada	salida
salida = referencia	0	nula	se mantiene
salida ≠ referencia	negativo	negativa	disminuye
salida ≠ referencia	positivo	positiva	aumenta

Cuadro 1.

Acción correctora de la retroalimentación negativa.

En cualquier organización donde tiene lugar una transformación, hay entradas y salidas. Las entradas resultan de la influencia del medio sobre los sistemas, y las salidas de la acción del sistema sobre el entorno, donde los dos están separados temporalmente.

Los sistemas viables precisan asegurar que la salida de la transformación no se aleje de lo esperado, normalmente contenida en los objetivos que el sistema asume como siendo aquello que determine su conducta en un estado operacional definido como de equilibrio interno o equilibrio fisiológico, dado, por su parte, por la constancia de su medio interno.

LA RETROALIMENTACION POSITIVA.

No todos los mecanismos de retroalimentación generan acciones que tienden a la mantención de una posición de equilibrio. Existe la retroalimentación positiva, que genera un comportamiento divergente. Esto es, frente a una determinada tendencia, de la conducta real, este lazo refuerza la tendencia. Como dice Rosnay "el mas arrastra al mas, hay un efecto de bola de nieve, del mismo modo en que el menos arrastra al menos", (ROSNAY-1977). Tal como ilustra la figura 7.

El crecimiento demográfico, los procesos de depresión económica, son ejemplos de retroalimentación positiva. En las dos situaciones, si el lazo actúa progresivamente, puede ocurrir la destrucción del sistema, sea por explosión o por minimización creciente de sus funciones. Por lo tanto, los lazos positivos deben ser controlados necesariamente por lazos negativos para que los sistemas puedan conservarse en el tiempo.

Inclusive, puede constituirse en un generador aleatorio de "variedad", aumentando las divergencias, y la complejidad, al multiplicar las posibilidades de interacción, aumentando las posibilidades de elección y acentuando la diferenciación.

Cabe resaltar que el crecimiento de un sistema depende de lazos de retroalimentación positiva, ya que éste al actuar siempre en el mismo sentido, conduce al aumento acelerado de ciertos valores, como el número de individuos en el crecimiento de la población; la diversidad en la variedad de los elementos que constituyen el sistema; la energía, como ocurre con los excedentes energéticos; y la acumulación de capital.

Las características fundamentales de la retroalimentación positiva son:

- Acentuar la diferenciación
- Multiplicar las posibilidades de interacción
- Aumentar el efecto de la transformación
- Determinar comportamientos divergentes
- Manifestar un efecto acumulativo o acelerar la transformación en el mismo sentido
- Facilitar la adaptación sistema-medio
- Permitir comportamiento innovador
- Actuar como generador aleatorio de variedad

Las conductas esperadas hacia las cuales los sistemas viables normalmente tienden, de modo de mantener su equilibrio, se pueden modificar. Los sistemas viables se adaptan

habitualmente a las condiciones variables del entorno, poseen detectores y comparadores para captar señales internas o externas y contrastarlas con valores de equilibrio.

Se se producen divergencias, se emiten señales de error para corregirlas. El sistema procura, en el caso de no poder alcanzar su anterior estado de equilibrio, nuevos puntos de equilibrio y nuevos estados estacionarios, a través del juego complementado de los lazos positivos y negativos.

Como dice J. de Rosnay "... La evolución de un sistema abierto es la integral de estas mudanzas y de estas adaptaciones, (ROSNAY-1977). Esta evolución, se manifiesta a través de niveles estratificados de organización y por la necesidad de propiedades nuevas.

Algunos ejemplos de evolución a niveles crecientes de complejidad son la evolución social y la evolución biológica. En cada nivel aparecen nuevas propiedades. Hay, por lo tanto, un salto cualitativo.

SISTEMAS DE CONTROL EN CIRCUITO ABIERTO.

Los sistemas de control de circuito abierto, son aquellos en los cuales la acción del control es independiente de la salida. Se trata de sistemas no retroalimentadores. Son los mas simples y los menos onerosos, aunque imprecisos y poco confiables, ya que el ajuste de control depende fundamentalmente del criterio y cálculo humano, además de su incapacidad de adaptarse a las variaciones de las condiciones del medio o alteraciones exteriores. Por ejemplo, en una lavadora no automática el tiempo de lavar depende del criterio y cálculo de la persona que la usa.

Um ejemplo aclarará esta situación.

Imagínese que la temperatura de una sala deba permanecer constante, con un valor previamente fijado. Se sabe que la temperatura de la sala depende de la temperatura exterior, del calor aportado por un calentador y posiblemente, de otros factores, como del abrir y cerrar de puertas y ventanas y de la presencia de personas en el ambiente.

Tratando el asunto bajo el prisma de control, se tiene, tal vez, algo como lo que muestra la figura 8. La temperatura de la sala se modifica cuando la temperatura exterior se altera; en particular, la temperatura ambiente baja cuando está frío en la región externa. Consecuentemente, el ambiente precisa ser calentado si cae la temperatura exterior. El aumento de calor puede ser obtenido, por ejemplo, por medio de un calentador. En este contexto la tarea del control es hacer que ese calentador produzca mas calor, si la temperatura exterior baja. La relación explícita de dependencia entre la temperatura externa y la abertura que se debe dar al calentador es posible de determinación, estableciendo una cierta proporcionalidad

transformando uno de los valores en el otro mediante un adecuado equipamiento mecánico o eléctrico. En suma, la temperatura ambiente sube automáticamente, si la externa cae, y baja automáticamente, si la externa sube. De esta manera, la temperatura de la sala se mantiene fija, con valor constante.

Es claro que el control, en vez de realizarse automáticamente, puede llevarse a cabo por una persona. En este caso, simplemente, se trata de control manual.

A partir de aquel ejemplo, haciendo abstracciones, se llega a un esquema de circuito de control, en general representada por un diagrama de bloques (figura 9).

Hay control cuando una situación Z_2 (la situación bajo control) se ve influenciada por la mensuración de otra situación Z_1 (la situación a controlar) y por el tipo de relación que se establece entre Z_1 y Z_2 . Una forma de control se dá, por ejemplo, cuando la temperatura de una sala aparece influenciada por la temperatura exterior. Cuando la iluminación de las calles de una ciudad se fija en base a las estaciones -las luces se encienden mas temprano en el invierno y mas tarde en verano. En un control (al contrario de lo que acontece en la regulación) la intervención se dá en función del valor que se influencia. Rasgo típico de todo control es la forma abierta en que ocurre el paso de Z_1 a Z_2 . La desventaja de cada control reside en que está sujeta a perturbaciones no previstas con antelación. Perturbaciones imprevistas son aquellas que no quedan determinadas por medio de relaciones físicas generales y que, por eso, no pueden ser contrabalanceadas por mecanismos de graduación como, por ejemplo, el hecho de abrir una ventana.

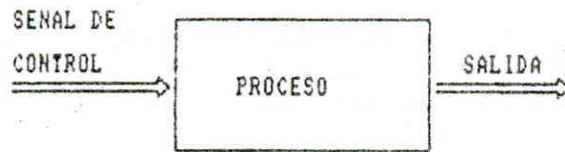


Figura 1

MODELO BASICO DE CONTROL

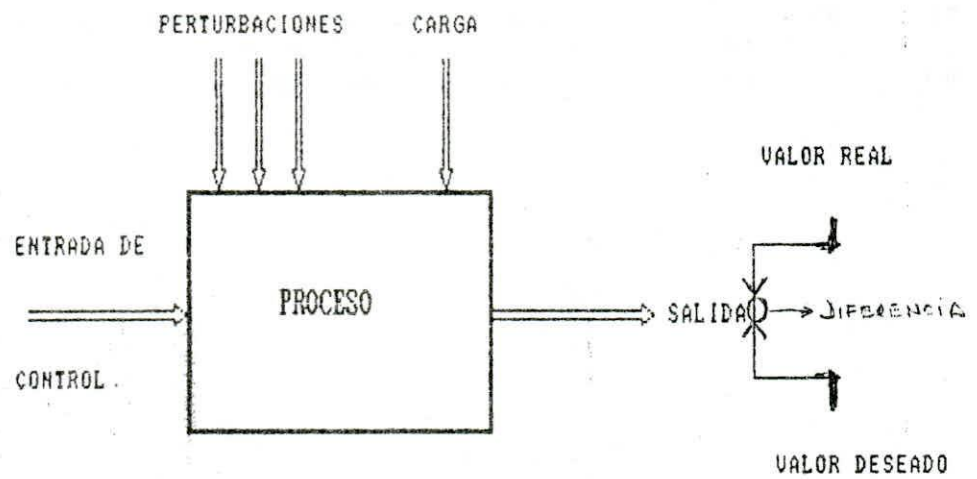


Figura 2

SISTEMA DE CONTROL DE CIRCUITO ABIERTO.

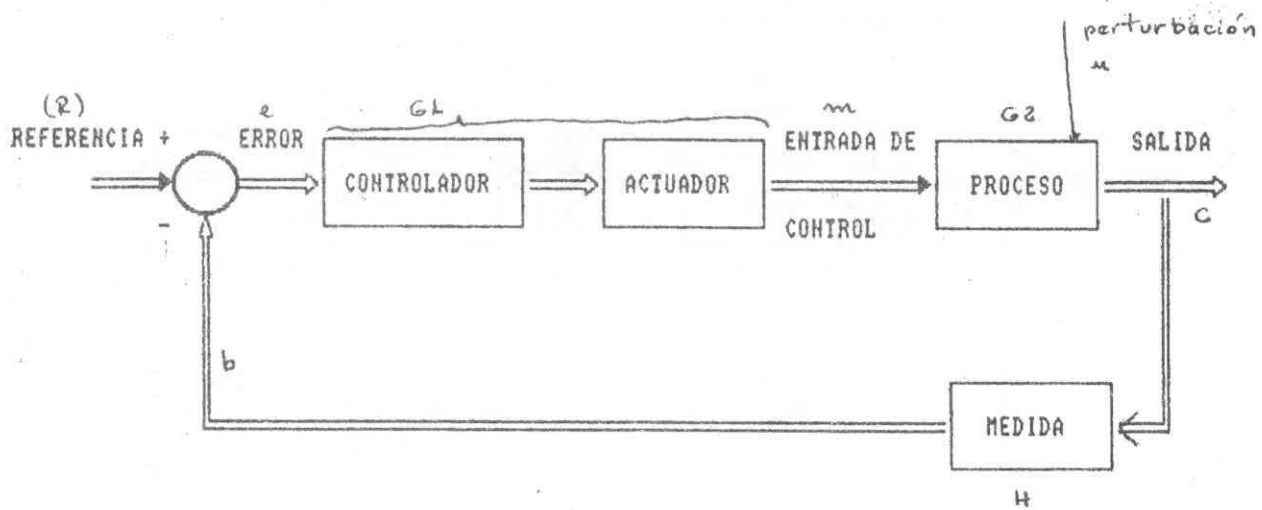


Figura 3

SISTEMA DE CONTROL EN CIRCUITO CERRADO.

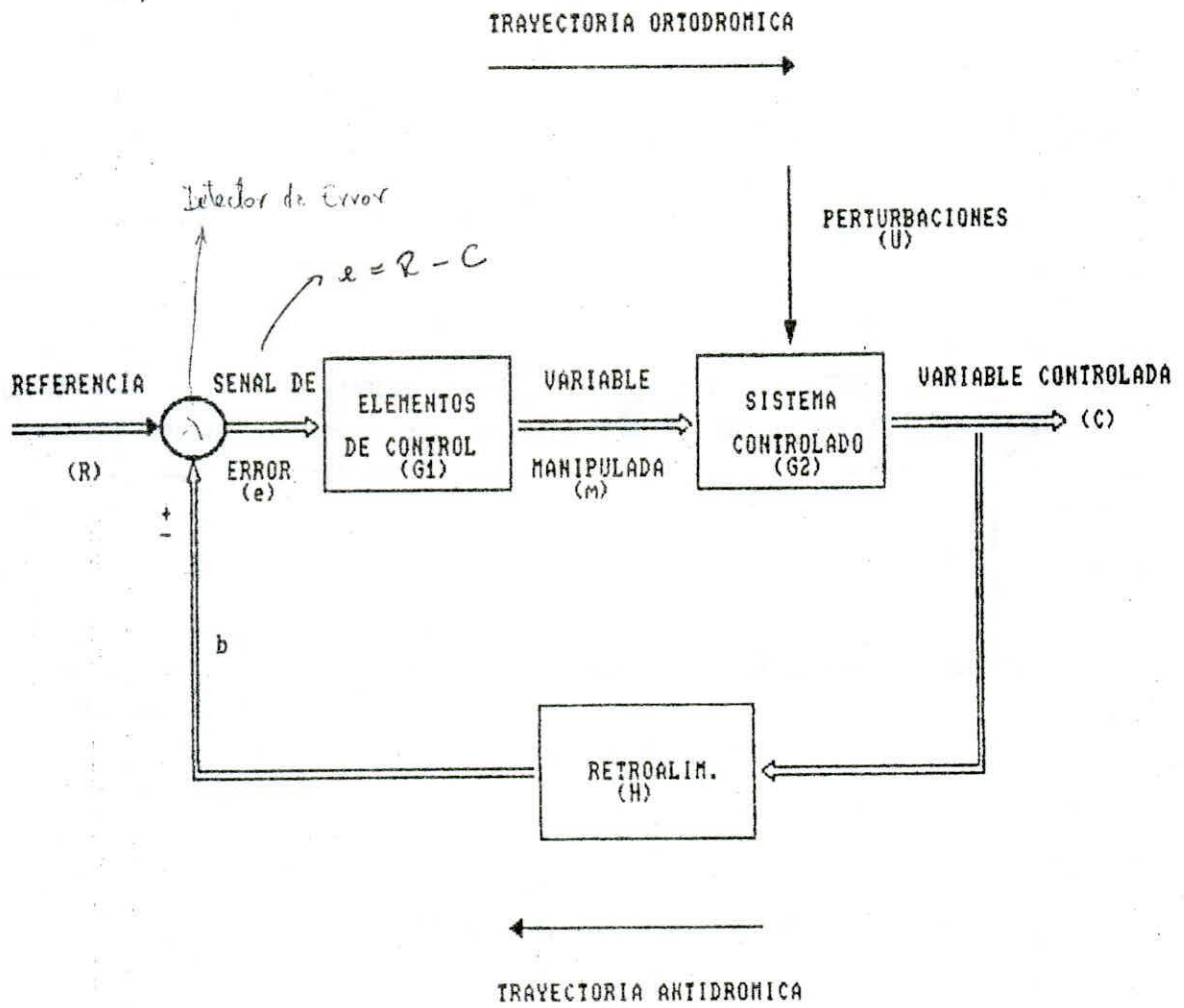


Figura 4

ELEMENTOS DE UN SISTEMA DE CONTROL EN CIRCUITO CERRADO.

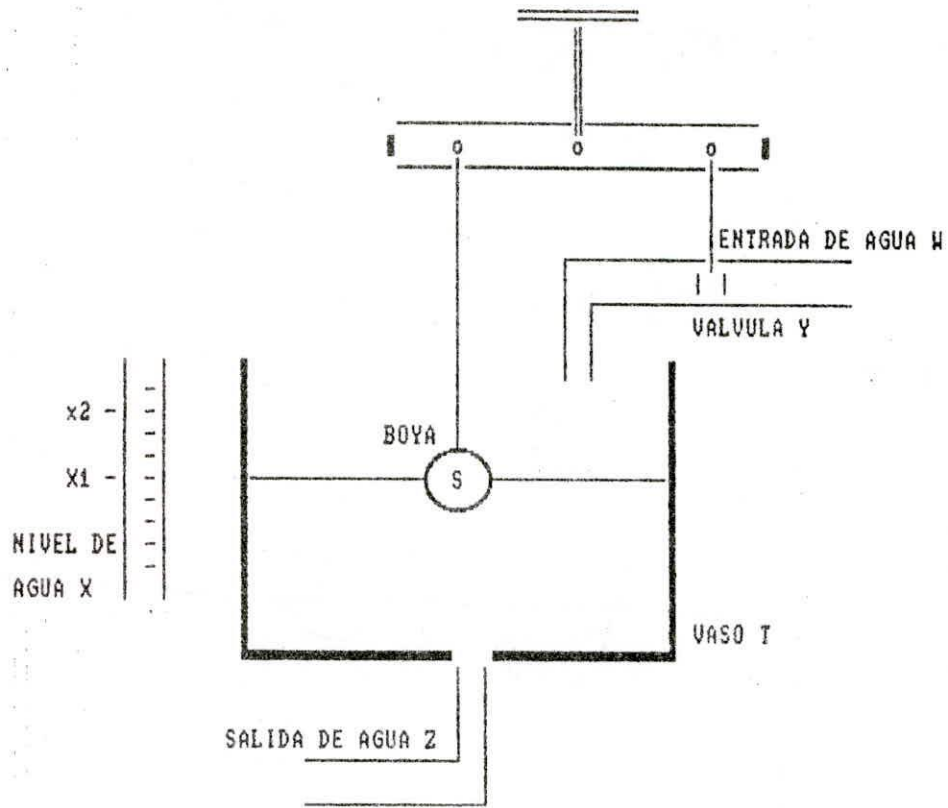


Figura 5

REGULADOR DE ESCORAMENTO.

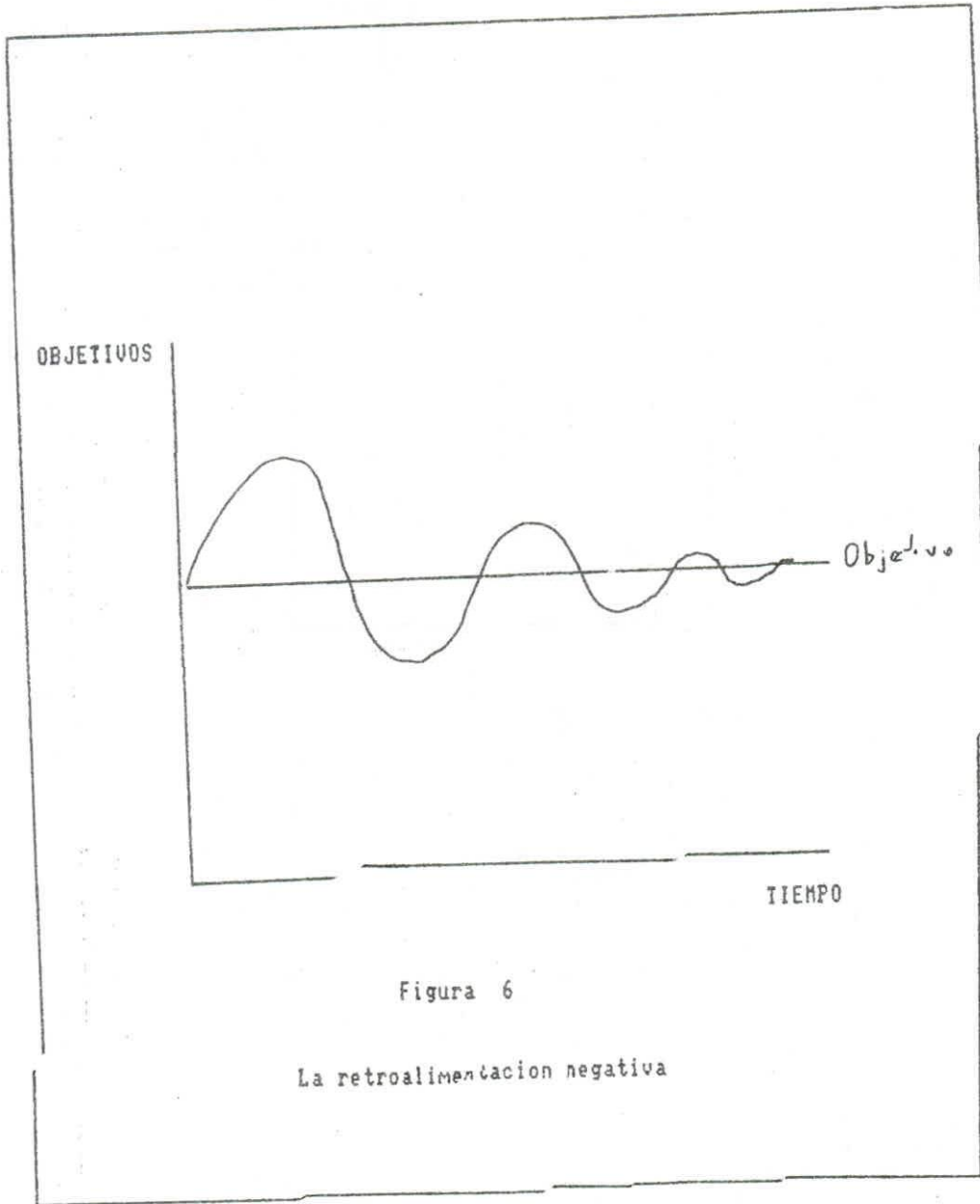


Figura 6

La retroalimentacion negativa

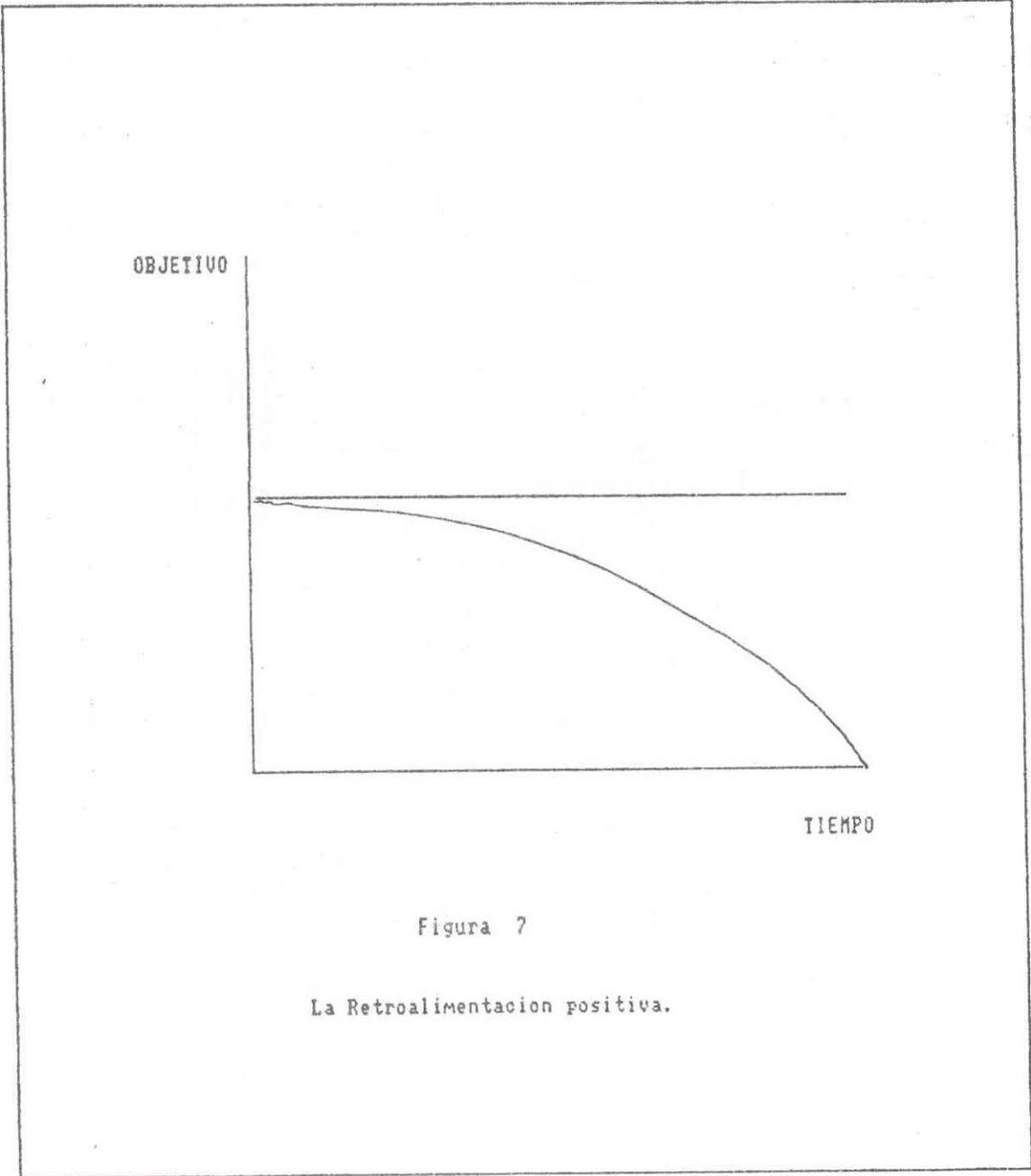


Figura 7

La Retroalimentacion positiva.

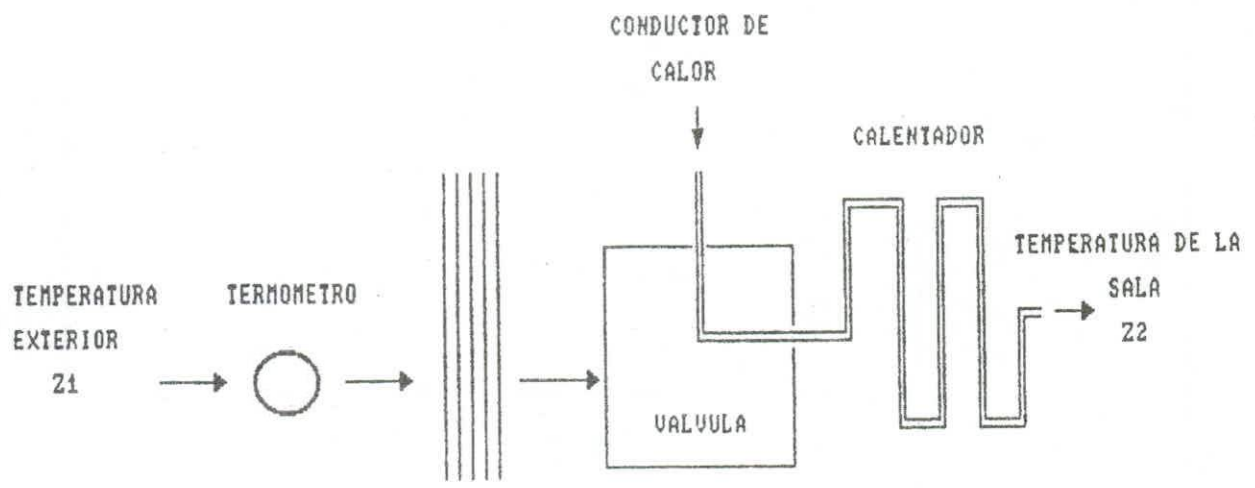


Figura 8

CONTROL DE TEMPERATURA DE UHA SALA.

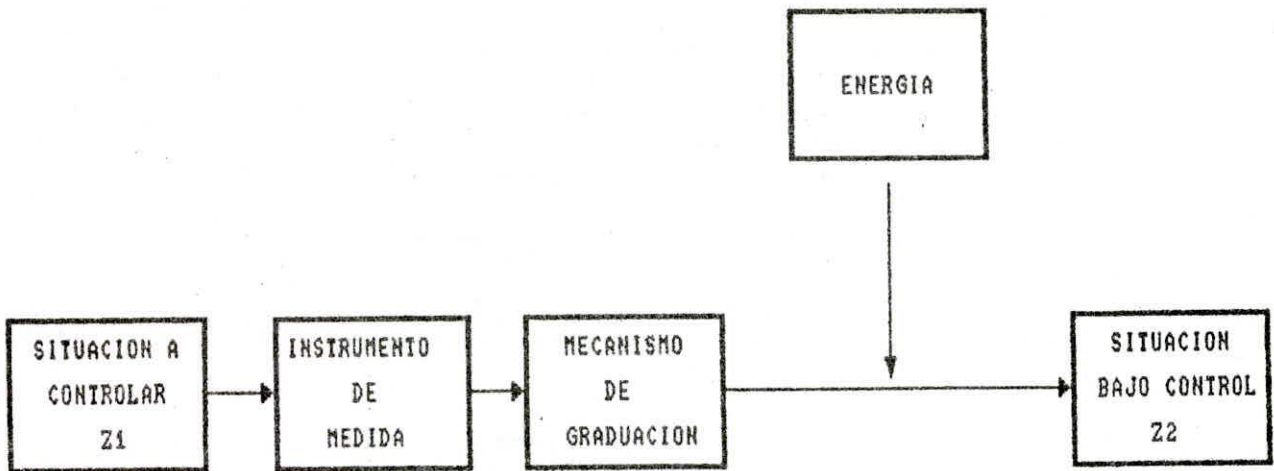


Figura 9

REPRESENTACION DE UNA CADENA DE CONTROL ABIERTA.

LA HOMEOSTASIS Y LOS FACTORES DE CONTROL.

Asegurar la mantención de las organizaciones de carácter productivo, y por lo tanto, su capacidad efectiva de desarrollar una actividad que tiende a alcanzar ciertos objetivos, exige que sus procesos, a partir de las coherencias operacionales propias de estas organizaciones, presenten como una de sus características fundamentales el carácter homeostático de su comportamiento, lo que se traduce, en la mantención constante de las variables que expresan la conducta de los procesos. De esta forma la propia estructura del sistema conformada por una red de componentes estratificados actúa, a través de mudanzas estructurales, sobre sus relaciones con el medio externo y enfrenta cualquier acción en ella generada en ese medio.

Claude Bernard citado por Pierre Vendryes, (VENDRYES-1969), al referirse a los organismos biológicos, ya en 1850, llama a aquellos mecanismos de mecanismos compensadores.

Por lo tanto son estos mecanismos compensadores los que tienen que permitir el equilibrio interno -en función de su medio ambiente inmediato- tanto del organismo, en cuanto sistema, como de sus componentes estructurales. Estos compensadores operan en función de los valores de determinadas variables de control a través de las cuales se expresa el comportamiento organizacional. A través de estas es posible asegurar permanentemente el estado de equilibrio interno o equilibrio fisiológico de los componentes, permitiendo de este modo la normalidad operacional que el organismo como un todo requiere.

Esta operación puede ocurrir de diversas formas:

- a través de la compensación de los desvíos entre los valores esperados y reales de las variables de control.
- a través de la compensación de las perturbaciones. Esto ocurre, por ejemplo, en un sistema de control de la temperatura, cuando se compensa los desvíos de la temperatura de una habitación, en relación a los cambios que ocurren en la temperatura exterior.
- a través de la eliminación de las perturbaciones. En este caso, para mantener, por ejemplo, la temperatura de una habitación es preciso asegurar que todo cambio de la temperatura externa no influya sobre la temperatura interna. Algunos organismos vivos, para este caso disponen de caparazones (como la tortuga) como parte de su estructura. Esta elimina las perturbaciones que podrían ocasionar efectos no deseados.

Puede ocurrir también, que sea necesario modificar la conformación estructural del organismo, cuando las perturbaciones

son persistentes en su intención de provocar efectos no deseados.

De estas maneras es posible alcanzar aquel equilibrio del medio interno entendido como la base que proporciona la capacidad operacional óptima para alcanzar los fines a los cuales todo organismo o proceso orienta sus actividades, y que por la configuración estratificada de la organización de los sistemas viables le son entregados por un estrato sistémico- que se expresa también como organismo- de orden superior. En esta perspectiva estratificada alcanzarlos es, por lo tanto, la condición para su homeostasis operacional encunto organismo.

Al observar una organización estratificada es posible distinguir en ella un sistema, cuando es posible identificar una unidad compuesta es decir, una unidad simple en una primera distinción y los componentes en un distinción siguiente en términos tales que la composición de estos genera la unidad simple inicialmente distinguida. Visto el sistema desde el estrato inferior, puede decirse que la unidad simple es una totalidad emergente con características propias en relación a los elementos.

Así, el sistema existe como una unidad interna de sus elementos y como una unidad externa frente al medio. Esto es, el sistema puede ser observado bajo dos ángulos diferentes: internamente como una pluralidad de elementos, y externamente como algo separado, unitario, como un corte hecho en la realidad. Al mismo tiempo, los objetos considerados elementos relevantes para el sistema, tienen también como elementos del sistema una doble face. Estos elementos constituyen el medio interno del sistema -en un sentido horizontal- pero en lo que dice relación con la composición vertical del sistema, presentan una face hacia el sistema en el sentido ascendente y otra en sentido descendente. Para el sistema aparecen como elementos, pero para los niveles inferiores, cualitativamente diferentes al sistema del cual forman parte, ellos mismos aparecen como sistemas o constituyen sistemas.

Por lo tanto, aunque en un sistema los componentes o partes son considerados elementos, esto es, unidades indivisibles, estos elementos pueden ser, por su vez, considerados sistemas, al distinguirse en ellos una unidad compuesta.

Los componentes organizacionales involucrados en la homeostasis.

Todo organismo, a partir de la constancia de su medio interno desenvuelve su trabajo para alcanzar determinados objetivos, que corresponden a la compatibilidad procesal requerida por la participación en instancias -sistemas- organizacionales mayores. Cuando en ese contexto se hace necesario realizar una acción de verificación de la efectividad y de la eficiencia alcanzada, además de la verificación del equilibrio del medio interno, (para en virtud de eventuales desajustes entre lo esperado y lo real, establecer

las medidas correctivas que sean necesarias), surge la necesidad de disponer de mecanismos homeostáticos.

Por lo tanto bajo una perspectiva estructural, conforme a las coherencias operacionales que determinan una organización de actividad humana, es posible distinguir dos sub-procesos, que aunque siendo indisolublemente complementarios, aparecen como necesariamente distinguibles. Uno de estos corresponde a los "procesos operacionales" o "proceso primario", el otro a los "procesos de control".

El primero tiene como misión la realización de las tareas necesarias para la obtención de los objetivos entregados al organismo. Pertenece al dominio de las transformaciones del tipo insumo-producto y su contenido es operacional.

El segundo consiste en la dirección del sub-sistema que realiza las transformaciones, para asegurar el logro de los objetivos en eficiencia y eficacia -a partir de la efectiva constancia del medio interno- razón por la cual su trabajo es esencialmente decisonal.

Esto quiere decir que en la estructura organizacional en cada proceso -organismo- en el cual se realizan determinadas transformaciones "insumo-producto", estas deben estar ligadas a la instancia de control que deben garantizar su propia homeostasis encunto sistema, mas también, la homeostasis operacional organismo-medio, asociada a la relación con los otros procesos, a través del ajuste de su comportamiento real a su comportamiento esperado. Este último pre-establecido anteriormente, bajo la forma de objetivos de actividad o parámetros fisiológicos.

Esta instancia de control debe operar a dos niveles sistémicos y en ellos debe, tras aceptar los objetivos originados en su medio, en el primer nivel, "activar" lo controlado y en el segundo, "ajustar" su conduta cuando su comportamiento se aleja de ciertos valores esperados, tanto en lo referente al alcance de los objetivos de actividad como en relación a la mantención de los equilibrios fisiológicos internos requeridos.

La "activación" consiste en (a partir de los objetivos del sistema) entregar los sub-objetivos derivados de aquellos a los procesos que conforman el sistema controlado. El "ajuste" consiste en hacer compatible el comportamiento real de los sub-sistemas con su comportamiento esperado, establecido en función de aquellos objetivos y en mantener las condiciones fisiológicas internas que la óptima operacionalidad requiere.

Por lo tanto, la instancia de control "activa" el proceso con todos sus sub-procesos y también lo "ajusta" encunto sub-proceso que ha recibido un sub-objetivo entregado por un nivel sistémico inmediatamente superior.

Cuando "activa", distribuye sub-objetivos a cada sub-proceso

del sistema, de modo tal que las instancias de control de cada uno de estos tiene que ajustar el comportamiento de la transformación que realiza en función de aquellos sub-objetivos entregados. Así, se puede decir que la tarea del control se lleva cabo a dos niveles, presentándose recurrentemente a través de toda la estructura estratificada de la organización, como aparece en la figura 1.

Las expresiones del control en las estructuras estratificadas.

Dado el carácter estratificado de las estructuras de los organismos viables, en todo sub-sistema de un sistema mayor, que tenga la misión de alcanzar determinados objetivos distribuidos por aquel, se realiza una tarea de control, supeditada a la instancia que le entrega aquellos objetivos. Por lo tanto, percibido este sub-sistema en función de dos estratos sistémicos: sub-sistema e instancia inmediatamente superior; el control opera a dos niveles: un primero en el cual se determina y distribuye los objetivos y un segundo en el cual se reciben estos objetivos. Se establece, así, el carácter, el contenido y los alcances de las actividades operacionales del tipo entrada-salida que el organismo debe realizar para cumplir con su misión organizacional.

Para distinguir entre aquellas dos formas a través de las cuales se expresa el control, la primera de ellas se denominará "Control" y la segunda "Regulación", cualquiera que sea el estrato donde este fenómeno es distinguido.

En esa doble dimensión, el "Control" corresponde a la instancia que a través de un proceso de activación determina el ámbito de las tareas de control -mas allá de darle inicio- fijando los parámetros de actividad reflejados en los objetivos que establece. La regulación es aquella dimensión del control que debe operar a partir de aquellos parámetros ajustando las conductas de las actividades operacionales del organismo a los valores esperados contenidos en los parámetros. No poseen, por lo tanto, atribuciones para modificarlos.

En una estructura organizacional estratificada, conformada por diferentes procesos, cada uno de estos será considerado un sistema cualquiera que sea el estrato en cual se observe. En este caso, por proceso, o sub-proceso, se debe entender aquel sistema en el cual se distingue un "sub-sistema Regulador" y un "sub-sistema regulado".

Dado el carácter estratificado del organismo, se tiene un proceso P1, en un nivel sistémico cualquiera de la estructura, un regulador R1 controlando un sub-sistema regulado r1, y si el sub-sistema r1 se percibe como sistema, esto es, si se distinguen sus elementos, sean P21 P22, que también se configuran como sistemas, entonces en un segundo nivel sistémico se tendrá la configuración de la figura 2.

De esta forma, cualquiera que sea el proceso, en una

perspectiva homeostática, se puede describir exhaustivamente en dos niveles sistémicos, y dada la estratificación de las estructuras organizacionales, estas representaciones son recurrentes en la verticalidad organizacional.

Hechas estas consideraciones y para distinguir entre la "tarea de control" y la "instancia de Control" -aquella que realiza o materializa el control- se empleará "control" y "Control" respectivamente.

Por lo tanto, visto un organismo -encuanto proceso componente de un sistema mayor- a un nivel, el control -función de control-, se materializa a través de un Regulador actuando sobre lo controlado o regulado, esto es un sub-proceso o sub-función orgánica.

En una primera aproximación, estas dos instancias, se pueden representar como en la figura 3.

Aquel Regulador al recibir determinados sub-objetivos derivados de un estrato inmediatamente superior y asumir la responsabilidad de lograrlos efectivamente, tiene que mantener el equilibrio homeostático operacional entre el sub-sistema que controla esto es, o sub-proceso, o sub-función, y su medio inmediato, para lo cual debe también mantener el equilibrio del medio interno o equilibrio fisiológico.

Esto es, en última instancia el sub-sistema estará bajo control, si tiene capacidad de alcanzar tales objetivos, y si consigue realmente la correspondencia que se requiere entre ellos y las salidas de la transformación.

A otra instancia del proceso organizacional, el sub-sistema que debe ser controlado (la transformación) es aquella que realiza las transformaciones entrada-salida. Encuanto sistema está estructurado en un conjunto de sub-transformaciones (ST), donde cada una de estas tiene que realizar una tarea específica establecida a través de la entrega de objetivos cuyo origen, desde la perspectiva del proceso como un todo, es el estrato inmediatamente superior. De este modo, cada una de las sub-transformaciones realiza una tarea específica dentro del conjunto establecido para obtener la salida que se requiere de la transformación. Esta operación corresponde a la activación de la transformación y ocurre como presenta la figura 4.

Como cada uno de estos objetivos -sub-objetivos- tiene que ser alcanzado, cada sub-transformación precisa también de una instancia de control -un Regulador- que comande la transformación entrada-salida. Vale destacar que, visto ahora el proceso a dos niveles, el estrato inmediatamente superior corresponde al "Control", del mismo modo que el Regulador de cada sub-transformación, en su propio estrato es también "Control" en relación a los Reguladores de las sub-transformaciones que conforman la transformación -sub-transformación- que controla. Por lo tanto cuando en una estructura estratificada expresada a

dos niveles, aparecen dos instancias de control, una superpuesta a la otra, la instancia superior es "Control" y la instancia inferior "Regulador".

Las relaciones regulador-regulado.

Esta desagregación del control hacia las transformaciones explicita las relaciones Regulador-regulado. En estas, el problema fundamental reside en la necesidad que tiene el Regulador de conocer el valor de las variables de estado a través de las cuales se expresa el comportamiento de lo regulado (la transformación). Se trata específicamente de un problema de tratamiento de información, esto es, de la recolección de datos relativos a aquellas variables de estado y su posterior transformación en la información que requiere el Regulador.

Para eso el Regulador, que tiene que asegurar que la transformación alcance sus objetivos y mantener las variables fisiológicas en niveles normales, debe definir un conjunto de variables de control. Estas variables corresponden al lazo de retroalimentación que permite comprobar la efectividad de la transformación y su equilibrio fisiológico, materializando de este modo la homeostasis del sistema.

De esta forma, el equilibrio de la transformación, esto es, el equilibrio homeostático del sistema, se observará a través de los valores asumidos por las variables de control.

LAS VARIABLES DE CONTROL.

Para que el regulador pueda realizar una tarea efectiva, debe disponer de los instrumentos necesarios para ello. Estos corresponden a las "variables de control". Las variables de control son la expresión de los criterios que el regulador establece para configurar sus "cuadros" de control, para, así, poder asegurar el logro de los objetivos asignados al sistema en eficiencia y eficacia a partir de la necesaria constancia del medio interno que la capacidad operacional del sistema requiere. En el marco de la ley del requisito de variedad de Ashby, aquellas variables se configuran dada la necesidad de equilibrar la variedad en la relación regulador-regulado.

Estas variables, en una primera aproximación corresponden a dos tipos: "variables de control operacional" y "variables de control esenciales".

Las primeras corresponden a los criterios de eficiencia y eficacia que el logro de los objetivos requiere sean alcanzados.

Las segundas a aquellos factores que tienen relación próxima con la supervivencia, es decir, aquellos cuyo equilibrio garantice la constancia del medio interno, el cual asegura, a su vez, el nivel requerido de capacidad potencial para realizar las tareas necesarias.

Las "variables de control operacional" (vc) se agrupan en torno a lo que se puede denominar "vectores de control operacional", (VCO), cada uno de los cuales está compuesto de un conjunto de "variables de control operacional".

Por lo tanto, cada "vector de control operacional" (VCO) se puede interpretar de la siguiente forma:

$$VCO_j = (vc_1, vc_2, \dots, vc_i, \dots, vc_{n_j})$$

donde

VCO_j = j-ésimo vector de control operacional

vc_i = i-ésima variable de control operacional

n_j = número de variables de control operacional componentes del j-ésimo vector.

Lo mismo ocurre con las "variables esenciales". Estas se agrupan en torno a "vectores esenciales" (VE), con cada uno de los cuales también compuesto de un conjunto de "variables esenciales" (ve).

Cada vector esencial tiene entonces la siguiente composición:

$$VE_j = (ve_1, ve_2, \dots, ve_i, \dots, ve_{n_j})$$

donde

VE_j = j-ésimo vector esencial

ve_i = i-ésima variable esencial

n_j = número de variables esenciales componentes del j-ésimo vector.

Estas dos expresiones de las variables de control se expresan a través de valores esperados y reales.

En el caso de las variables de control operacional, los valores esperados; se relacionan con los objetivos a ser alcanzados por la transformación, razón por la cual deben estar implícitos en los objetivos que el proceso recibe. Operacionalmente representan el punto de partida para evaluar el grado de ajuste que a ellos manifiesta la transformación. Del mismo modo, dado que los procesos deben mantener el equilibrio del medio interno, que es aquel que asegura la mantención de la capacidad de realizar trabajo, se hace necesario conservar en su valor normal - esperado- un conjunto de variables "fisiológicas" que son aquellas que expresan la normalidad de la capacidad operacional

de la transformación.

Los valores reales, por su parte, provienen de los eventos que ocurren en la transformación. Son en consecuencia, expresiones de las variables de estado que expresan la conducta de la transformación.

Con estas dos expresiones de las variables de control es posible para el regulador determinar la dimensión de eventuales desajustes y, en función de estos, aplicar las acciones correctivas necesarias para asegurar el alcance de los objetivos entregados al sistema en eficiencia y eficacia, además del equilibrio de su medio interno, efectuando de este modo la homeostasis del sistema.

Estas variables de control son los instrumentos que permiten comprobar la efectividad de cada sub-transformación, por lo cual el equilibrio de la transformación, y por lo tanto el equilibrio homeostático del sistema -el proceso-, se puede observar y manejar a través de los valores que asumen dichas variables.

Los vectores esenciales y los vectores de control operacional, son por lo tanto, instrumentos de regulación, y estructuralmente, relacionan los nodos de "conducción" -o regulación- y "transformación". Aquella relación se manifiesta en la entrega de la información que requiere la regulación para ajustar las transformaciones, apoyando de este modo, el logro de los objetivos, mediatizada por el tratamiento a que se someten los datos asociados a los eventos que ocurren en la transformación.

Cabe destacar que, dado que los valores reales de estas variables de control tienen su origen en las variables de estado que expresan el comportamiento de la transformación, corresponden al flujo de información que circula por los conductos que, a su vez, están conformados por el conjunto de variables de control definidas por el regulador.

Aún más, es a través de estos flujos que se materializa la retroalimentación que permite corregir las desviaciones que se pueden producir entre la conducta esperada y la conducta real observada en la transformación, como muestra la figura 5. ✓

La información así generada y transmitida desde la transformación, alimenta las decisiones que deben tomar las instancias de regulación, lo cual hace que aquella información sea el componente esencial del circuito de retroalimentación.

La tarea de regular la conducta de la transformación -inserta en la acción de control del estrato inmediatamente superior- sólo es posible en la medida en que se disponga de la información relativa al rendimiento de la transformación, de modo de poder comparar lo real con lo esperado.

La información del comportamiento real es provista por las variables de control. Por lo tanto, estas corresponden a los

factores que es preciso medir para asegurar el logro de los objetivos y son determinados por la instancia de regulación, es decir por aquel que "conduce" la transformación. Para operar en efectividad con estas variables de control es necesario el uso y por lo tanto la implementación previa de instrumentos de medición, registro, transmisión y tratamiento de datos, los que deben permitir dar cuenta del comportamiento del medio interno y del alcance de los objetivos. El conjunto de instrumentos que permite que esto ocurra son denominados sistemas de información.

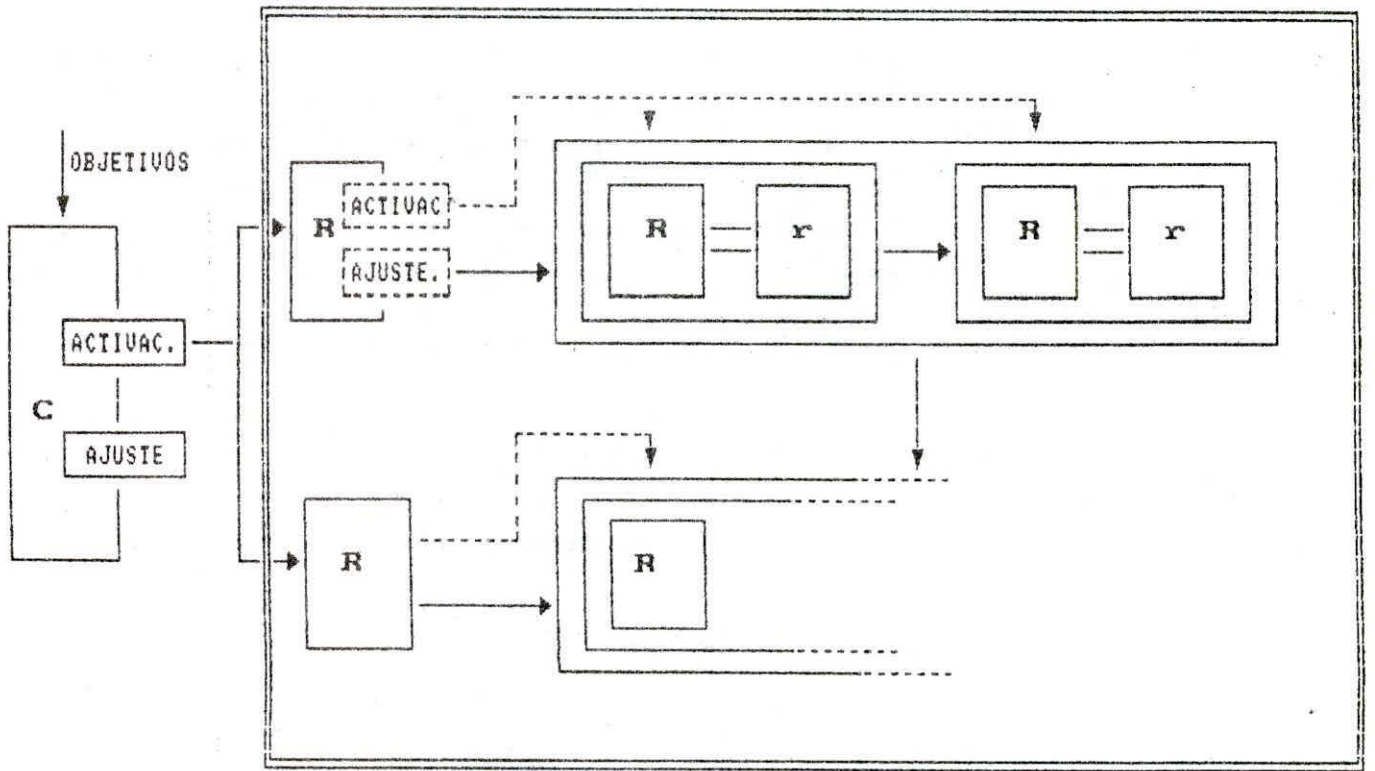


FIGURA 1

ACTIVACION DE LO REGULADO

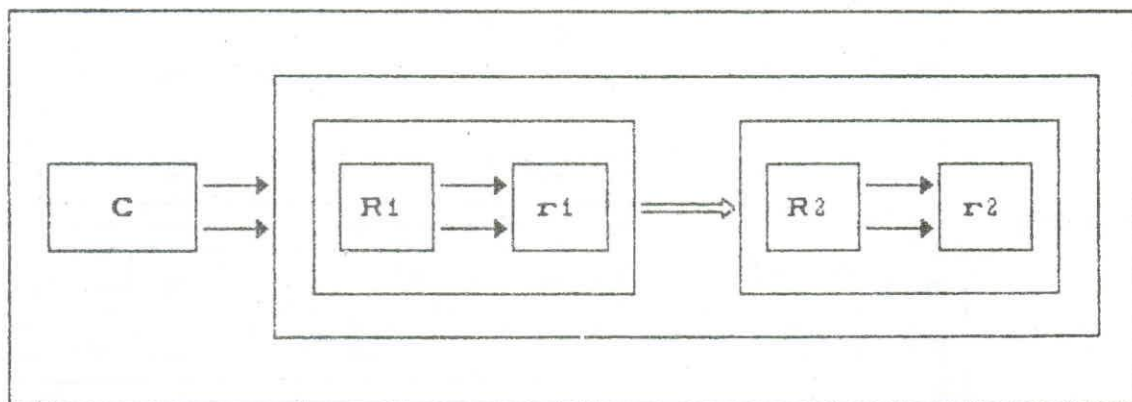


FIGURA 3

LAS DOS INSTANCIAS DE CONTROL.

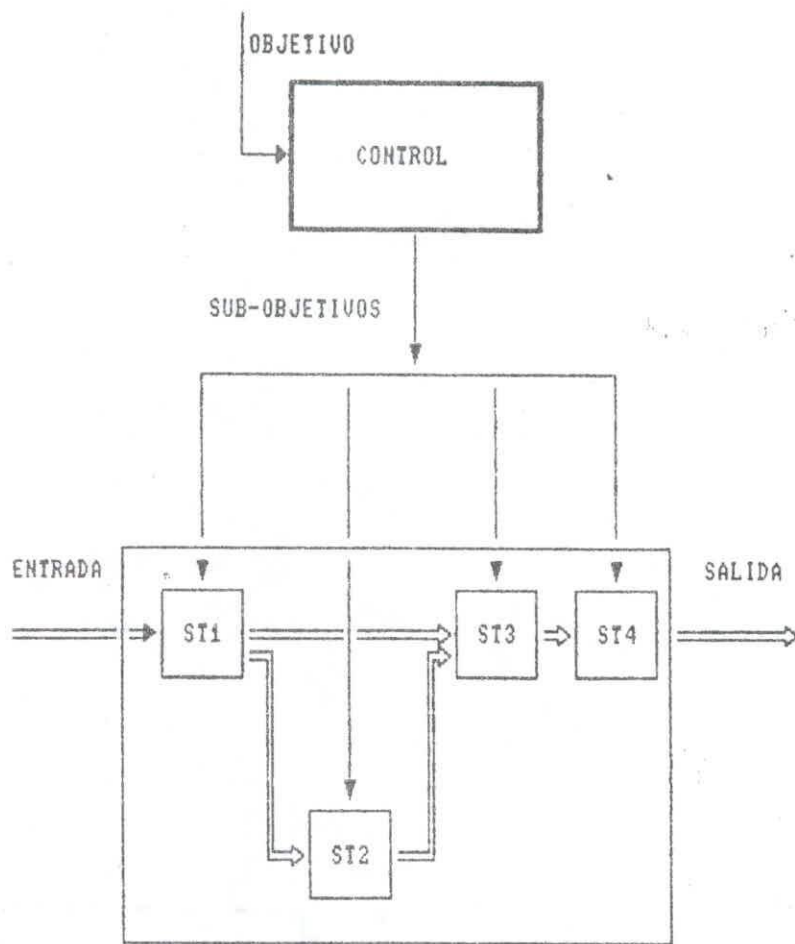


Figura 4

EL REGULADOR COMO CONTROL

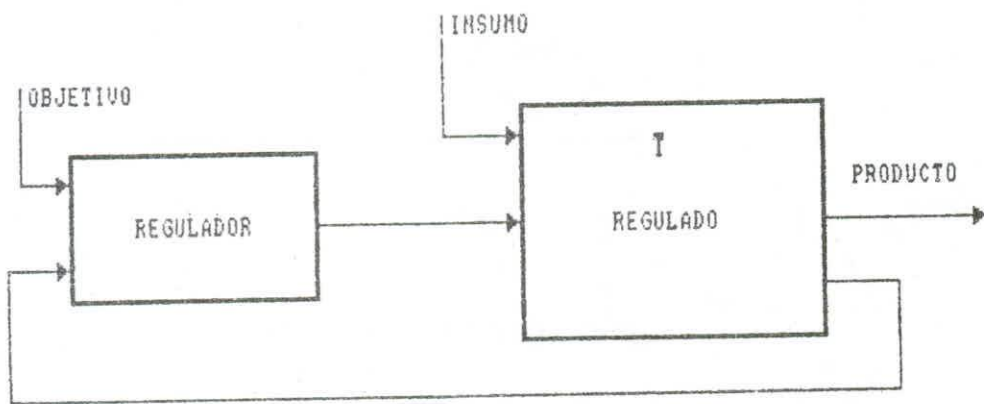


Figura 5

Relacion control-transformacion

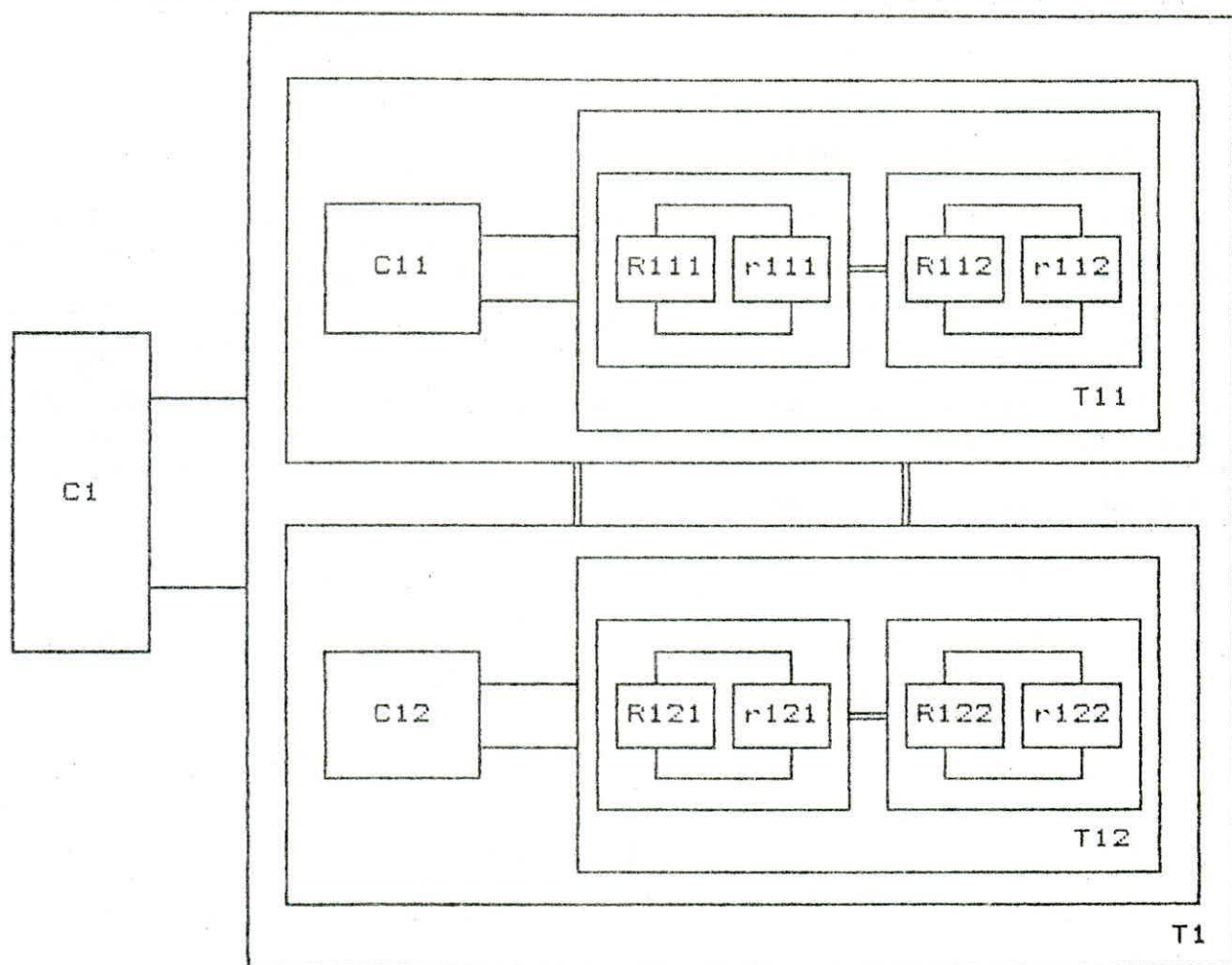


Figura 2 : Los roles de control y regulación a distintos niveles jerárquicos.

En la figura se puede observar que C11 y C12 en el tercer estrato son instancias de control, No obstante, en el segundo estrato son Reguladores, nivel en el cual C1 es Control.

En el cuarto estrato ^{R111 R112 R21 R22} R11, R12, R21 y R22, son Reguladores. Sin embargo, en el quinto estrato son instancias de Control.

Esto es, el Regulador y el Control, en cuanto instancias orgánicas, dependen del lugar que ocupan dentro de la estructura estratificada del sistema.

En los dos casos es necesario verificar el valor de las

variables asociadas a la variabilidad del proceso, para asegurar su equilibrio fisiológico y la eficiencia y eficacia de su acción. Por lo tanto, se puede afirmar que la información es el elemento fundamental en el proceso (de control y regulación.)

Esta doble dimensión de las tareas de control explicita tanto para el Control como para la Regulación algunas diferencias que es preciso destacar.

El control pretende la mantención del equilibrio homeostático entre el sistema y su medio. Esto sólo será posible en la medida en que la conducta del sistema sea compatible con su objetivo, el cual teniendo su origen en aquel medio, el sistema acepta como determinante de su acción.

Esto significa que el control persigue el dominio del sistema, considerando que este desarrolla su trabajo en un medio que por su alta variedad es turbulento y tiene la capacidad de afectar su viabilidad.

Para ello, una de las tareas fundamentales del control es establecer sub-objetivos que permitan alcanzar efectivamente tal equilibrio. Aquellos sub-objetivos son derivaciones del objetivo del sistema.

En el corto plazo el Control se expresa a través del proceso de regulación. En esta relación el Control debe rentener los valores de las normas del sistema. Aquellas, obviamente, pueden ser vectores. En este caso la regulación corresponde a la acción de compensar las desviaciones respecto de la norma cuyo valor está determinado por el control.

Puede decirse, por lo tanto, que el control es una función que se manifiesta en el largo plazo, entanto que la regulación lo hace en el corto plazo, en el escenario establecido por la acción de control.

Los mecanismos de regulación son simples homeostatos. Esto es, dispositivos que permiten mantener alguna variable dentro de los límites deseados, donde no necesariamente, como señala Stafford Beer, "el valor de este nivel debe ser invariante", (BEER-80).

Puede decirse, en consecuencia, que las relaciones Control-Regulación son relaciones estratificadas, donde el control siempre está en un nivel superior. A dos niveles lo que operacionaliza la actividad de Control es la Regulación.

EL PAPEL DE LA RETROALIMENTACION.

Uno de los canales de comunicación que conecta el Regulador y lo regulado es aquel que facilita el flujo de los valores de las variables de control referidos a los eventos que ocurren en las transformaciones que son controladas. Aquellos flujos son información potencial que pueden convertirse en información

en el momento de afectar los estados y la conducta de quien la recibe.

Cabe destacar que lo que se transmite no es información, esta se hace tal solo en la medida en que los valores de determinadas variables son recibidas por un receptor interesado.

En este sentido se tiene que entender la información cuando se afirma que es la instancia que al ser agente de disminución de la incertidumbre, afecta el comportamiento de quien la recibe

Cabe destacar que lo que se transmite no es información, esta se hace tal solo en la medida en que los valores de determinadas variables son recibidas por un receptor interesado.

En este sentido se tiene que entender la información cuando se afirma que es aquella instancia que (al ser agente de disminución de incertidumbre, afecta el comportamiento de quien la recibe. En este contexto, al interior de los procesos adquiere un sentido explícito toda vez que se constituye en un estímulo que generará un conocimiento efectivo de lo que ocurre en el sub-sistema regulado.

Así, conceptualmente, y considerando que la incertidumbre representa un estado entrópico, la información derivada de los valores de las variables de control que fluyen entre regulador y regulado, será básicamente neguentrópica al reducir aquellos niveles de incertidumbre.

No obstante, y dada la capacidad limitada de absorber variedad de los reguladores y considerando que la variedad de lo regulado es siempre mayor que la del Regulador, por las necesidades de eficiencia estructural del proceso, la información para ser funcional a la pretensión de efectividad del Regulador, no debe superar aquella capacidad limitada. Si esto ocurre se puede convertir en instancia generadora de entropía.

De esta forma, el valor de mínima entropía corresponde a aquel nivel donde la capacidad de absorber variedad del Regulador alcanza un máximo, con una máxima disminución de la incertidumbre. Por lo tanto la información que recibe del Regulador debe ser tal que además de disminuir la incertidumbre, conforme a la "Ley de requisito de variedad de Ashby", no ultrapasa su capacidad de absorber variedad.

Aquella información potencial, asociada a las variables de control, fluye por canales de retroalimentación.

En al) configuración estructural, Regulador-regulado-retroalimentación, los reguladores dependen para tener éxito, de dos artificios esenciales.

- el primero es una comparación continua -automática- de los valores reales y esperados de las variables de control.

- el segundo es una retracción automática -o inmediata- de las medidas correctivas.

Los reguladores -gerencias en las organizaciones de carácter productivo- normalmente están conscientes de la importancia de estos artificios y hacen considerables esfuerzos para su utilización. Pero no alcanzan un monitoreo continuo ni una corrección inmediata. Entonces para dotar a las variables de control de valores reales, de modo que el proceso bajo control, pueda materializar la regulación de manera efectiva, es necesario substituir los modelos hierárquicos de transferencia de control por modelos homeostáticos de modo tal que el proceso disponga de determinados sub-sistemas para el tratamiento de la información. Estos deben realizar las tareas que van desde la recolección de los datos ligados a los eventos que ocurren en la transformación regulada, y que se expresan a través de los valores de sus variables de estado, hasta la obtención de información posible de generar en la medida en que las variables de control asumen sus valores.

Obviamente la efectividad de las medidas correctivas dependerá de la exactitud de los datos recogidos, del tiempo de respuesta del sistema regulador y de la calidad de la respuesta del Regulador y de la aceptación y -respuesta- a las medidas correctivas por parte de lo regulado.

Podría ocurrir que un determinado Regulador no sea capaz de mantener el equilibrio de la transformación que regula, colocando en peligro el equilibrio del proceso del cual forma parte. Si esto ocurre debe intervenir la instancia de control que actuando como Regulador, frente al conjunto de sub-procesos que conforman el proceso que controla, visto ahora como un todo, opera sobre las variables de control agregadas respectivas.

LOS SISTEMAS DE INFORMACION.

Esta estratificación de las variables de control solo es posible en la medida en que entre Regulador y regulado exista un instrumento de control que opere sobre aquellas variables. Esta es la única forma de pasar de un estrato a otro superior, considerando lo que ocurre en los niveles inferiores. De esta forma, los valores de las variables de control en los niveles inferiores, mas allá de satisfacer los requerimientos del Regulador respectivo, pueden satisfacer aquellas de la instancia de control inmediatamente superior.

Esto quiere decir que aquellos instrumentos de control - susceptibles de ser denominados sistemas de información- también deben responder al carácter estratificado de las organizaciones.

La información potencial generada por ellos alimenta determinados procesos decisórios que residen en los procesos de control. En ellos se compara lo que realmente ocurre en la transformación con determinados valores esperados. De aquellos procesos decisórios emanan las acciones correctivas que se

la verticalidad de la Organización, asociando nodos tecnológicos a los sistemas de información, para obtener, así, una estructura estratificada a través de nodos informáticos.

La importancia de distinguir entre los sistemas tecnológicos y los sistemas de información, reside en que no es suficiente incorporar dispositivos tecnológicos. No se trata simplemente de adicionar computadores a los reguladores, o configurar redes, ya que además de operar en tiempo real los sistemas de información deben tratar de evitar cubrir al regulador con información excesiva, situación esta que puede limitar la efectividad homeostática. Para eso el diseño de los sistemas informáticos debe considerar los aspectos esenciales del control y el rol de las variables de control en la operatividad efectiva de las tareas de los reguladores. Uno de estos problemas es la necesidad de equilibrar la variedad entre regulador y regulado.

Para enfrentar el problema de la variedad es preciso que las variables de control se expresen, sea como variables discretas - cantidad producida o personal en actividad, por ejemplo- como razones o como indicadores más complejos. Además de eso resulta conveniente expresarlos de manera gráfica

LA LEY DEL REQUISITO DE VARIEDAD DE ASHBY.

La regulación y el control, procesos esenciales para el mantenimiento del equilibrio homeostático del sistema con su entorno, se rigen por la ley de variedad de W. R. Ashby. Ashby, al abordar el problema de la regulación y el control, enuncia la denominada "Ley de la Variedad Requerida", fundamental para la teoría general de la regulación, como él mismo lo señala. Esta ley expresa que "sólo la variedad puede absorber la variedad", o dicho de otro modo y en las propias palabras de Ashby: "la variedad puede destruirse sólo por medio de la variedad".

En lo específico, este enunciado se puede interpretar señalando que cualquier solución que se pretenda dar a un problema complejo no puede ser menos compleja en relación a aquél, si lo que se pretende es, justamente, solucionar el problema. En otras palabras, no puede haber soluciones simples a problemas complejos. De no ser así, las soluciones serán parciales o irresolutivas.

De acuerdo a la ley de variedad de Ashby, si se tiene un sistema a controlar o regular, llamado S, y un sistema de control o regulación, llamado R, entonces el sistema R controlará o regulará efectivamente al sistema S, si y sólo si, la variedad de R es mayor o igual a la variedad de S, es decir habrá un control - o regulación - efectivo sólo si

$$\text{variedad (R)} \geq \text{variedad (S)}$$

Si se tiene, por ejemplo, un supuesto agregado de cuatro elementos sin relaciones entre ellos, denominado S, cuya variedad estará dada por:

$$\text{variedad (S)} = \log_2 (4) = 2 \text{ [bits]},$$

entonces cualquier instancia R que pretenda regular a aquél deberá tener, a lo menos, la misma variedad. Una solución obvia para diseñar este sistema R se obtiene al resolver la inecuación:

$$\text{variedad (R)} \geq \text{variedad (S)}$$

$$\log_2 (\text{número de elementos de R}) \geq 2$$

$$\text{número de elementos de R} \geq 2^2$$

$$\text{número de elementos de R} \geq 4,$$

basta entonces, tomar el número de elementos de R igual a cuatro para establecer la relación regulador-regulado según lo muestra

la figura 1.

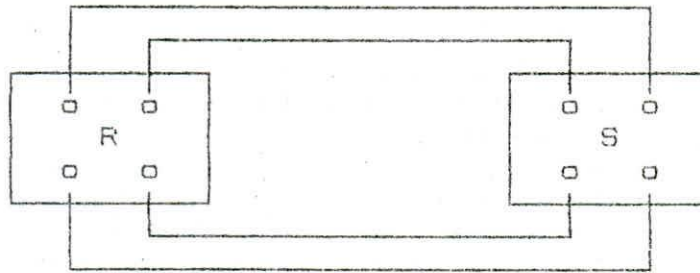


Figura 1: Relación regulador-regulado para los sistemas R y S.

Sin embargo esta situación, aunque eficaz, es ineficiente, por cuanto se estaría reproduciendo cuantitativamente en el sistema regulador R el sistema regulado S.

Dado que los sistemas además de ser eficaces deben ser eficientes, su diseño se establece en términos tales que estructuralmente, el sistema regulador manifiesta una menor variedad que el sistema regulado. Pero ello viola la ley de variedad de Ashby, por lo cual, para no perder la eficacia del control en aras de la eficiencia, es preciso establecer mecanismos para conceder al sistema regulador la variedad requerida para cumplir con esa ley. Ello se logra aportando flujos de información que relacionan ambos sistemas. La información aportada debe ser tal que la instancia de regulación o control reconozca los alcances de los eventos que se suscitan al interior del sistema regulado o controlado, y provea al regulador - o controlador - de los instrumentos de dirección que le permita la conducción o guía del sistema regulado - o controlado -, cuya acción también se materializa a nivel de flujos de información.

En otras palabras, si en un sistema compuesto por dos subsistemas, las variedades están desequilibradas, no puede asegurarse la estabilidad. Si es el regulador el que dispone de menos variedad hay sólo dos maneras de satisfacer la exigencia de la ley de Ashby:

- una es atenuar la variedad producida por el sub-sistema regulado, y
- la otra, amplificar la variedad del regulador.

entonces la condición para las amplificaciones y atenuaciones queda

$$\sum_i (AR_i) + \sum_i (aSi) \geq VS - VR \quad (1)$$

es decir, el aporte de todas las amplificaciones sobre VR y de todas las atenuaciones sobre VS, debe superar la diferencia de variedades entre los sistemas S y R, si se pretende que el sistema R regule eficazmente al sistema S.

En este caso, de la figura 4:

$$\sum_i (AR_i) = AR_c + AR_o, \text{ y}$$

$$\sum_i (aSi) = aS_c + aS_o$$

Por tanto, reemplazando en (1) se tiene

$$AR_c + AR_o + aS_c + aS_o \geq VS - VR$$

$$\implies VR + AR_c + AR_o \geq VS - (aS_c + aS_o) \quad (2)$$

Si ahora se define

$$VR' := VR + AR_c + AR_o, \text{ y}$$

$$VS' := VS - aS_c - aS_o,$$

como las variedades de los sistemas R y S, respectivamente, al incorporar los instrumentos de amplificación y atenuación, se tiene finalmente de (2)

$$VR' \geq VS'$$

que corresponde a la formulación de la ley de variedad requerida de Ashby.

Se puede concluir entonces, que la información es el elemento fundamental en el proceso de control y regulación.

Obviamente, estas dos estrategias se pueden combinar tal como lo muestra la figura 2.

En consecuencia, se puede afirmar que entre el sistema regulado y el sistema regulador se establecen determinadas relaciones que corresponden a dos categorías distinguibles:

- una de ellas es la transmisión de información operacional,
- la otra es la transmisión de información de comando.

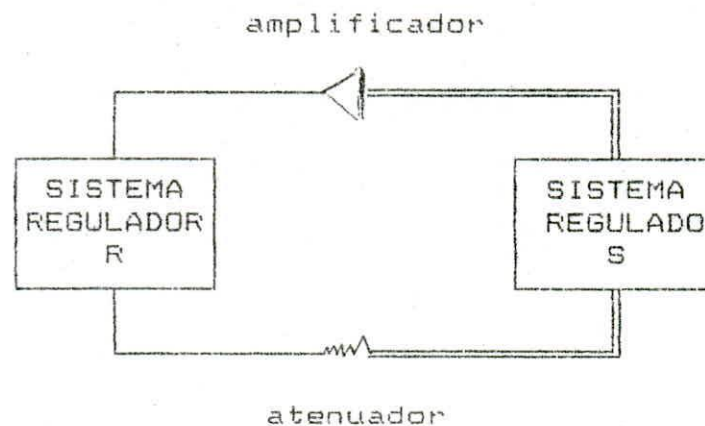


Figura 2: Estrategias para satisfacer la ley de variedad de Ashby entre los sistemas R y S.

Estas dos categorías de relaciones pueden distinguirse en la figura 3.

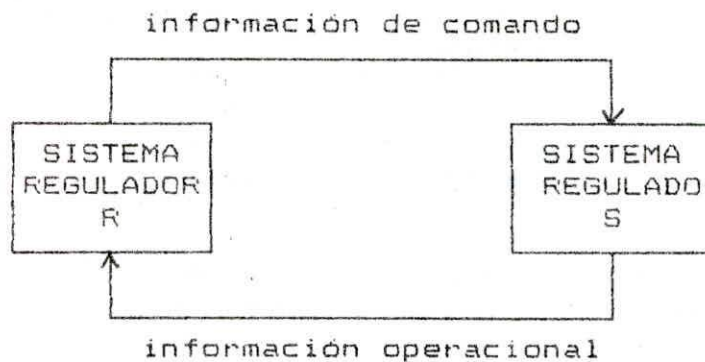


Figura 3: Flujos de información entre los sistemas R y S.

En virtud de la primera, el sistema regulador adquiere la noción del comportamiento real del sistema regulado, a través del valor de las variables en función de las cuales se determina su conducta.

La segunda, es aquella mediante la cual el sistema de regulación conduce el sistema regulado.

Se puede decir entonces, que entre el sistema regulado y el sistema regulador - o el sistema controlado y el sistema controlador - existen dos canales por los cuales fluye información de dos categorías diferentes.

Asociados a cada uno de ellos existen ciertos instrumentos para enfrentar el problema del desequilibrio de variedad, que se suscita por los requerimientos de eficiencia entre el sistema que regula y el sistema que es regulado, y que para un resultado efectivo requiere que se cumpla con la ley de variedad de Ashby. Estos instrumentos actúan como atenuadores y/o amplificadores de variedad.

En relación a la información operacional que comunica a ambos sistemas:

- existe un atenuador de variedad, es decir un instrumento que permite reducir la variedad que fluye del sistema regulado al sistema regulador, y
- a la par, aquel instrumento se constituye en un

amplificador de la variedad del regulador.

Este instrumento, en las Organizaciones humanas, se conoce como Sistema de Información.

Por su parte, en relación a la información de comando que maneja el sistema regulador:

- existen instrumentos que permiten amplificar la variedad que fluye desde el sistema que regula al sistema regulado,
- a la par, ese mismo instrumento atenúa la variedad del sistema regulado.

La información operacional entendida como la transformada semántica de los datos que reflejan lo que acontece en el sistema regulado, y que tiene la capacidad potencial de hacer disminuir la incertidumbre de quien la recibe, es un atenuador de variedad, ya que al no referenciar todos y cada uno de los eventos que se suscitan en la transformación, y que se registran en un primer momento como datos, está, por su carácter globalizante, inhibiendo la transmisión de toda la variedad que manifiesta el sistema que se está regulando.

Ello queda claramente de manifiesto por ejemplo, en los estudios de mercado, en donde lo que se tiene al final es información agregada, pero no corresponde a una relación biunívoca con los sucesos que se suscitan en aquella realidad que se está estudiando.

Los indicadores, tasas, ratios y otros, también representan atenuadores de información. Si se tiene por ejemplo, el valor del ingreso per cápita, éste en cuanto promedio, no expresa el valor del ingreso real de cada uno de los habitantes del país para el cual fue determinado.

Pero la información también es un amplificador de la variedad del sistema que regula, puesto que por medio de ella puede tener noción de lo que sucede al interior del sistema a regular, es decir, a través de ella puede incorporar variedad a su capacidad de regular, lo cual ocurre cada vez que disminuye su incertidumbre.

En relación a la información de comando, el regulador dispone siempre de ciertos niveles de autoridad con la cual puede

asumir la conducción de lo regulado. Esta autoridad es el instrumento a través del cual determina ciertas normas, reglamentos y procedimientos en virtud de los cuales el sistema regulado debe desarrollar su quehacer. Por lo tanto, mediante la autoridad de que se dispone, el regulador puede atenuar la variedad del sistema regulado al inhibir algunas expresiones de conducta, pero al mismo tiempo, amplifica su variedad puesto que le permite disponer de instrumentos para enfrentar la mayor complejidad del sistema que regula, ello ocurre por ejemplo, cuando tiene la opción de aplicar mecanismos de castigo-recompensa.

Por lo tanto, en la relación que se establece entre ambos sistemas, se incorporarán ciertos instrumentos de amplificación y atenuación de variedad asociados a los sistemas de comando y los sistemas de información, respectivamente, tal como lo muestra la figura 4.

En la figura 4, puede identificarse los siguientes elementos, todos medidos en [bits]:

V_S = variedad del sistema S regulado por el sistema R,

V_R = variedad del sistema R regulador del sistema S,

a_{S0} = información operacional que atenúa la variedad del sistema S,

A_{R0} = información operacional que amplifica la variedad del sistema R,

A_{Rc} = información de comando que amplifica la variedad del sistema R, y

a_{Sc} = información de comando que atenúa la variedad del sistema S.

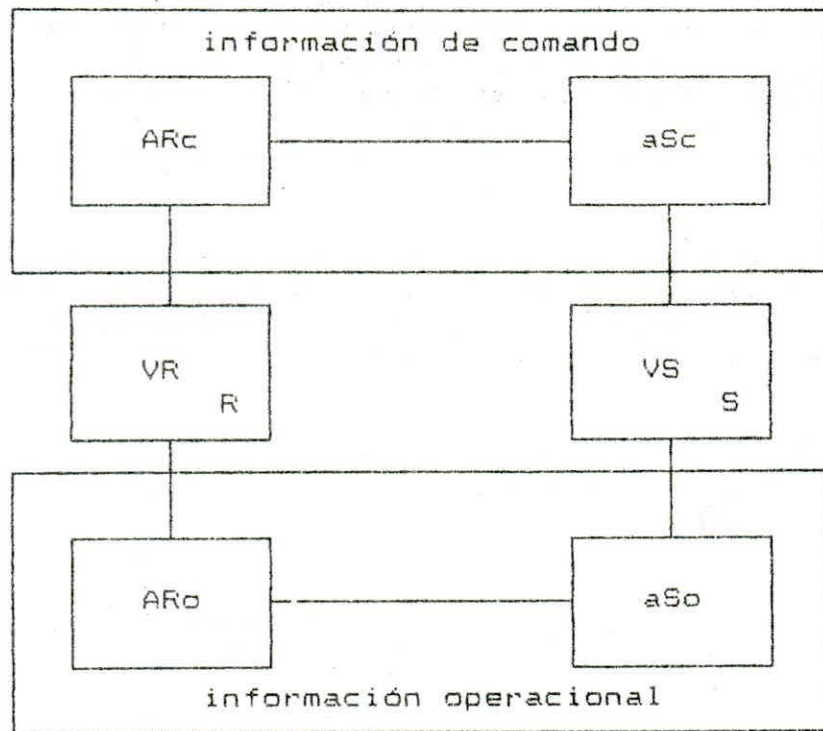


Figura 4: Instrumentos de amplificación y atenuación de la variedad.

Se quiere mostrar que, al incorporar los instrumentos de amplificación y atenuación, bajo ciertas condiciones, se cumple la ley de variedad de Ashby.

Se sabe que por razones de eficiencia

$$VS > VR \implies VS - VR > 0$$

es decir, $(VS - VR)$ es la variedad adicional que posee el sistema S por sobre la del sistema R.

Luego, si

AR = amplificación sobre VR, y

aS = atenuación sobre VS,

5. LA PROPUESTA DE STAFFORD BEER

CONTENIDOS

Introducción.

I. Componentes Conceptuales.

Conceptos y Términos.

Las Dimensiones del Problema.

La Organización de Sistemas Incomprensibles.

Jerarquías de Control.

II. La Forma del Modelo.

La Anatomía de la Administración.

Fisiología del Control.

Autonomía.

Administración Autónoma.

El Switch más Grande.

INTRODUCCION

Este trabajo describe los componentes conceptuales sobre los que se basa Stafford Beer para construir su modelo cibernético de organización.

Beer comienza planteándose tres puntos base en su estudio, que aunque data de la década del, sigue siendo válido en la actualidad para muchas organizaciones:

- Trata de captar la esencia de los problemas administrativos.
- Propone que el computador es una herramienta con la que se puede enfrentar dichos problemas, pero que no se ha entendido cómo usarlo para tal propósito.
- Debido a esto último, se necesita un nuevo enfoque para resolver el problema, el cual puede ser provisto por la Cibernética.

El trabajo se divide en dos capítulos, el primero de los cuales define el problema y estudia los elementos conceptuales del control para plantear una solución; el segundo capítulo construye un modelo de organización basado en el modelo del control en el cuerpo humano.

I. COMPONENTES CONCEPTUALES

En la actualidad se vive un profundo cambio tecnológico, pero la estructura de las organizaciones, que debiera ser acorde a ese cambio, se resisten a él, es decir, tienen problemas de adaptación.

Si es la razón del cambio en la tecnología la que coloca los problemas de adaptación, al parecer la única alternativa que queda es ir a la ciencia en búsqueda de la solución. Esto lleva consigo hacer una reconsideración de la organización tradicional.

La solución que se debe buscar no consiste en mejorar la organización actual, sino que se debe inventar una nueva forma de organización, aún de características desconocidas.

Hasta el momento, se ha tratado de asimilar el computador al modo de organización tradicional, pero la mayoría de los problemas subsisten. Esto se debe a que no se ha comprendido que el computador ofrece a la administración su propia tecnología.

En relación al problema planteado, debe entenderse que el administrador es el instrumento de cambio, lo cual significa que su tarea es de **control**, en el sentido que no es diseñador de sistemas de procesamiento de datos, sino diseñador de sistemas de control.

Así, el problema del control se sitúa como principal y es aquí donde surge la importancia de la Cibernética en la administración. En relación con esto, debe tenerse en cuenta que los principios fundamentales del control se aplican a todos los grandes sistemas, en particular a los administrativos. Además, bajo el enfoque que se está planteando, debe tenerse presente los dos siguientes aspectos:

- Hay que promover el estudio de la administración a un nivel donde esté relacionada con la naturaleza de las cosas y sus relaciones estructurales.
- El computador debe ser visto como algo más que una máquina de cálculo y usarlo de forma más inteligente. El computador es útil en el esfuerzo de control.

CONCEPTOS Y TERMINOS

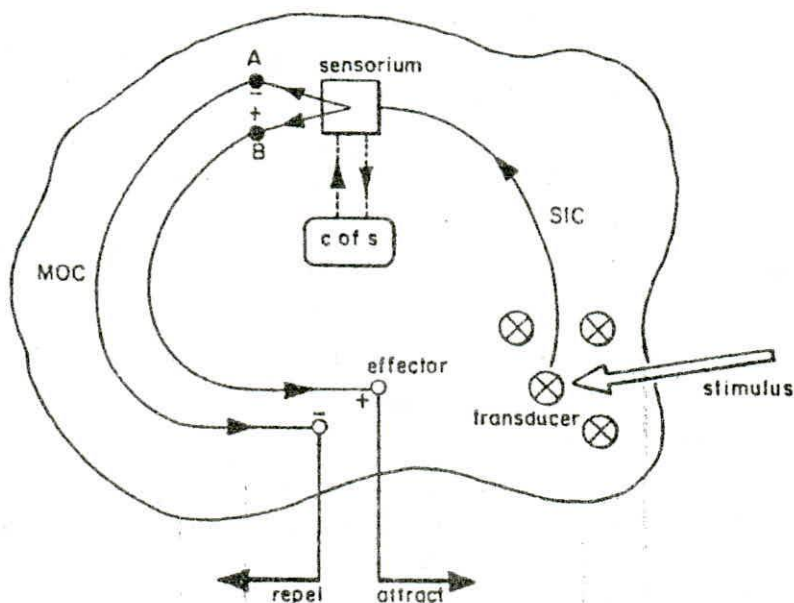
En esta sección se describirán aquellos elementos del control que serán útiles para comprender la solución que se planteará al problema que se está discutiendo.

- El primer principio del control es que el controlador es parte del sistema bajo control. Esto implica que el controlador crece y evoluciona con el sistema.
- Un **estímulo** es algo que altera la operación del sistema. La **respuesta** del sistema es algo que puede ser interpretado en términos del estímulo. Esto significa que el sistema evita un estímulo que interrumpe su actividad y busca incrementar un estímulo que favorece su actividad. Se dice que un sistema conforme a este comportamiento es **conocedor**.

El control facilita la existencia y la operación de los sistemas, en el sentido que la acción del control es una respuesta al cambio, una vez que el estímulo se ha registrado.

Es un error pensar que los sistemas pueden conocer cómo responder a aquellos estímulos conocidos y clasificados por adelantado. Entonces no es necesario que el sistema conozca por adelantado lo que puede causar un disturbio, sino captar que este está sucediendo y ser capaz de alterar los estados internos, hasta que los efectos del disturbio cesen. Un sistema que puede funcionar de esta manera se llama **ultraestable**.

Se describirá a continuación una versión simple del **controlador** de un sistema. Este se grafica de la siguiente forma:



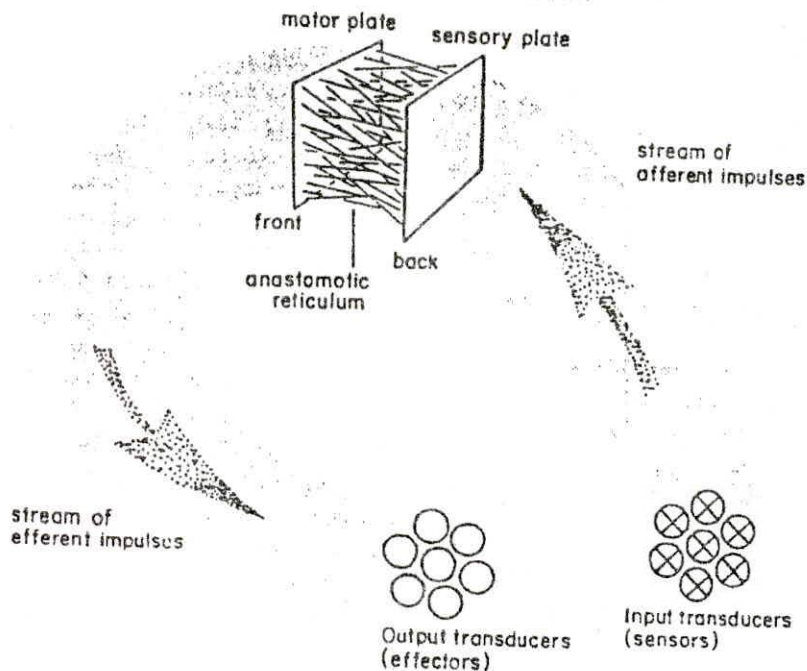
El **sensorium** es aquel elemento del sistema que puede registrar y clasificar la existencia de un estímulo. El controlador puede evitar (-) o reforzar (+) el estímulo; debe comparar las salidas con su criterio de estabilidad y su estrategia consiste en hacer pequeños aumentos o disminuciones, testeando los resultados del experimento. Si el experimento se extiende por mucho tiempo el sistema entrará en **oscilación**.

Hay un dispositivo, llamado **traductor**, que detecta el estímulo y lo ingresa al sistema en términos que tengan significado para el controlador.

El canal por el cual fluye el mensaje de entrada se llama **canal sensorial de entrada (SIC)**. El **canal motor de salida (MOC)** cierra el circuito y se llama así debido a que la salida tiene significado sólo en el caso que conduzca efectos. El motor de salida conduce a los **efectores**, que son capaces de accionar en relación al estímulo.

Las reglas que gobiernan cómo trabaja el controlador son conocidas como **función de transferencia $f(p)$** y su forma matemática es una ecuación diferencial. La acción del controlador es llamada **switcheo**, debido a que gatilla efectos como reacción a un estímulo.

En sistemas grandes y complejos, la circuitería de canales de entrada y de salida es muy complicada. Es posible que en un sistema no se pueda identificar los canales de entrada y de salida en forma individual, sino el grupo completo de ellos, lo que implica que no se puede identificar las funciones de transferencia individuales. Lo que sucede es que las entradas sensoriales llegan al sensor en forma diseminada, mientras los gatillos del motor de acción están también desplegados amplia y densamente en un plano completo, que corresponde a los puntos A y B del gráfico anterior, el cual puede extenderse de la siguiente forma:

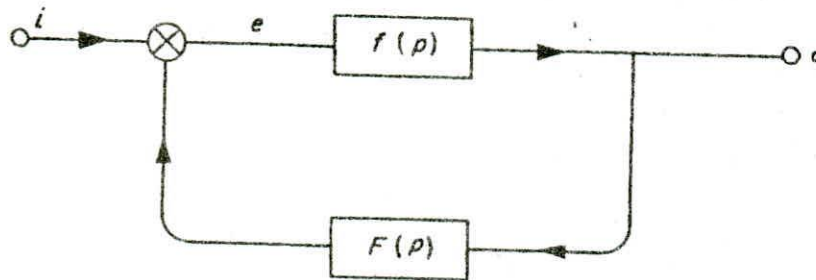


De esta forma, el problema del switcheo involucra conectar un lote completo de entradas y un lote completo de salidas. En lugar de un solo switch entre ellos, se tiene una red de conexiones, la cual es llamada **retículo** y debido a que su variedad satisfacen las leyes cibernéticas, se denominan **anastomótico**. De esta forma no se puede saber la fuente original de un estímulo.

Es importante destacar la noción de retículo anastomótico, debido a que los procesos de decisión dentro del cuerpo humano y dentro de las sociedades administrativas trabajan de esta forma, pues se puede observar la información que se está recibiendo y la acción que se está gatillando; se detecta los canales afectivos y efectivos por los cuales se realiza el trabajo.

El estímulo alerta una colonia completa de traductores de entrada y la respuesta del sistema es implementada a través de una colonia completa de traductores de salida (efectores).

La retroalimentación tiene su propia función de transferencia $F(p)$, la cual es necesaria para producir una disminución de las fluctuaciones de entrada. Dado que esto se puede hacer, se tiene como resultado un mecanismo de autoregulación que no se basa en las causas del disturbio, sino que trata con sus efectos en forma confiable. Esto se grafica de la siguiente forma:



Para obtener la función de transferencia total del sistema, se define una función $P(p)$, la cual subsume a $f(p)$ y a $F(p)$.

Es importante destacar que la retroalimentación negativa corrige la salida en relación a entradas fluctuantes, obteniéndose la propiedad sistémica normalmente deseable, la **ultraestabilidad**.

LAS DIMENSIONES DEL PROBLEMA.

Hay tres componentes fundamentales en cualquier sistema de control: entrada, salida y la red que los conecta.

Hay un arreglo de entrada, que comienza con un conjunto de receptores, que traducen información desde alguna situación externa a los canales efectivos y que concluye en un registro sensorial, en el cual se almacena esta información.

El canal que está transmitiendo debe tener capacidad suficiente, no sólo en la habilidad para distinguir las diferentes señales, sino también en cuanto a su velocidad.

Hay una placa del motor que transmite instrucciones a un conjunto de efectores que no pueden ser identificados entre ellos.

La tercera parte del sistema de control es el retículo anastomótico, que conecta el sensor con la placa del motor.

Según la Ley de Variedad Requerida de Ashby, la variedad de la salida debe a lo menos ser igual a la variedad de la entrada, tanto para el sistema como un todo como para los arreglos de entrada y de salida, considerados separadamente.

Situaciones de alta variedad son difíciles de manejar, debido a que la medida de la variedad es la medida de su incertidumbre. Una forma efectiva de manejar la variedad es a través de **elementos de decisión** binarios (sí/no) y esto es efectivo debido a que en muchos sistemas naturales, tales como el sistema social y el cuerpo humano, se toma decisiones de este tipo. De esta forma, la complejidad es medida en bits.

El problema de la proliferación de la variedad es muy complejo de manejar, pero debe tenerse presente que los sistemas naturales lo hacen con eficiencia. Ante esto, debe concluirse que un sistema debe enfrentar este problema con **organización**.

LA ORGANIZACION DE SISTEMAS INCOMPENSIBLES

Los sistemas bajo discusión son muy complejos para llegar a conocerlos a cabalidad y por este motivo se les llama incompensibles. En la naturaleza se encuentra muchos de estos sistemas que funcionan perfectamente, a pesar que no se puede determinar exactamente cómo lo hacen. Estos sistemas naturales tienen una capacidad inherente para auto-organizar el retículo anastomótico, pero no se puede entenderlo apropiadamente. Usan un mecanismo que evalúa todas las posibles salidas y escoge la mejor según algún criterio de eficiencia.

En relación a este último tema, se discutirá a continuación algunos aspectos importantes relacionados con los algoritmos y las heurísticas.

Un **algoritmo** es una técnica o un mecanismo que describe cómo alcanzar un objetivo, especificando cada paso completamente.

Una **heurística** especifica un método de comportamiento que tenderá hacia un objetivo que no puede ser especificado en forma precisa. Las heurísticas presentan reglas generales para alcanzar objetivos, pero no presentan una ruta exacta hacia un objetivo, como lo hace un algoritmo.

Estos dos conceptos son muy importantes en Cibernética, para tratar con sistemas incomprensibles, para los cuales normalmente es imposible presentar un algoritmo. Pero, por lo general, no es muy difícil presentar una clase de objetivos, lo cual permite pensar más en términos de heurísticas que de algoritmos como una forma eficiente para enfrentar el problema de la proliferación de la variedad.

En relación e esto último, se puede pensar que al utilizar el computador en la organización del retículo anastomótico, debe programárselo si se conoce el objetivo global, usando un algoritmo determinado por una heurística, de tal forma que esta última va generando una estrategia para irlo guiando en forma progresiva hacia el objetivo. Esta forma de control, que va aprendiendo a medida que avanza, es conocida como **control heurístico** y al respecto se puede mencionar los siguientes aspectos relevantes:

- La heurística puede comenzar a disminuir el peso que su estrategia de control da a su entrada, eliminando información fuente errónea, de tal forma que el resultado que entrega no depende estrictamente de la entrada.
- La estrategia generada por una heurística puede ser tan compleja que no puede ser entendida por el ser humano.
- Suponiendo que un sistema de control ha sido efectivo y que ha generado una estrategia que no se puede comprender, se concluye que debe existir otro sistema de control, que utilice un lenguaje de mayor orden y que use la salida del primer sistema como su entrada y producir nuevas salidas en otro plano. Pero este criterio de ganancia a su vez no puede ser especificado, por lo cual el segundo sistema necesita un tercer sistema para evaluar sus salidas en un lenguaje de alto orden. Este tercer sistema puede experimentar heurísticamente con el tiempo base de las evaluaciones económicas del segundo sistema.
- El argumento anterior continúa en una jerarquía de sistemas y los niveles de lenguajes que van con ellos buscan algún tipo de criterio último, el cual es la **supervivencia** del sistema.
- Cuando se dice que una heurística organiza un sistema para aprender, tratando una nueva variación de su estrategia operacional, también se debe decir que tal heurística organiza una familia de sistemas **para evolucionar**, tratándose de una nueva mutación en su estrategia de control genético. El objetivo de la **adaptación** es idéntico.

Las técnicas heurísticas son determinadas dentro de un esquema que especifica el modo, los límites y el criterio de búsqueda. Si el esquema es a sí mismo una heurística, entonces requiere otro esquema y así indefinidamente. En algún punto el n-ésimo esquema debe ser declarado como el esquema absoluto. Esto puede ser realizado en la práctica, pero no en la teoría. La mejor posibilidad para cambiar la dirección hacia una adaptación más exitosa radica en una reorganización de estas jerarquías de comandos.

Esto último hace ver la necesidad de utilizar el método de prueba-error, mediante el cual el sistema va aprendiendo, adaptando y evolucionando. Este método lleva a una retroalimentación necesaria para el cambio.

JERARQUIAS DE CONTROL

Dentro de un sistema de control existen elementos de decisión, llamados **nodos**, los cuales forman una red. Estos nodos están condicionados por un metasistema, que usa el método heurístico de prueba-error, que se llama **ciclo algedónico**. El conjunto total de estos elementos se llama **algedónodo**.

En un ciclo algedónico, que consiste de un algoritmo estipulando una heurística, el algoritmo es especificable solamente en un metalenguaje. Esto significa que el sistema de segundo orden es enlazado al primero mediante su propio ciclo algedónico. El proceso avanza, formando una jerarquía de comandos y puede hacerlo indefinidamente. En lógica teórica se demuestra que el sistema total requiere un número infinito de metalenguajes. En la práctica, se debe adoptar un último metasistema.

La conclusión anterior tiene paralelos en los hechos ordinarios de la vida institucional. En una empresa, los departamentos son coordinados en divisiones, las divisiones en grupos y los grupos en corporaciones. Los diversos niveles son autónomos y los controles son principalmente algedónicos. El gerente de la empresa mira hacia arriba a un metasistema llamado "la industria" y más allá otro llamado "la industria". Los dos están enlazados a esta empresa por ciclos algedónicos.

La forma de control que se está discutiendo indica que el control se constituye mediante estructuras jerárquicas, donde en cada nivel debe distinguirse los elementos de decisión y los comandos. Dos ejemplos de elementos de decisión son los administradores y las neuronas.

Cuando en un sistema comienza a haber acontecimientos, el elemento de decisión comienza a acumular evidencia y cuando está seguro de que hay evidencia suficiente para demandar acción, es decir, cuando la suma de las entradas excede un valor umbral preestablecido, gatilla.

Los sistemas viables revelan un retículo anastomótico en lugar de una red diseñada de elementos de decisión. Dentro del retículo, los elementos se forman y reforman a sí mismos en estructuras apropiadas. Estos elementos, que son los puntos nodales, son gobernados por funciones de transferencia, que cambian en forma probabilística más que de ecuaciones diferenciales.

Las retroalimentaciones no son simples dispositivos para corregir errores, que ajustan las salidas a valores correctos, sino que son nodos algedónicos que provienen de sistemas de alto orden.

II. LA FORMA DEL MODELO

Utilizando los elementos de Cibernética estudiados hasta el momento, se propone como objetivo construir un modelo de organización de cualquier sistema viable.

Para el propósito propuesto, debe entenderse los principios de viabilidad y se considera que para ello la mejor alternativa es estudiar un modelo viable conocido. Se escoge el cuerpo humano por ser tal vez el sistema viable más rico y flexible de todos.

Dentro de este desarrollo se comparará continuamente el desarrollo de la regulación corporativa del cuerpo humano con sus manifestaciones en la empresa, con el fin de verificar la validez del sistema escogido.

LA ANATOMIA DE LA ADMINISTRACION

La empresa, que es la entidad que controla el administrador, es un buen ejemplo de un sistema de alta complejidad en el cual la entrada y la salida son a sí mismos subsistemas de alta variedad. Lo que conecta la entrada a la salida es la empresa misma y la actividad compleja total se ejecuta en un retículo anastomótico.

Se desarrollará un diagrama organizacional que indica cómo cada parte de la empresa se relaciona con cada una de las otras partes, con el propósito principal de determinar dónde radican las responsabilidades. En los diagramas se mostrarán primero las partes principales y luego las partes secundarias y junto a ellos irán descripciones detalladas del trabajo que realizan, con el fin de mostrar cuál es el trabajo global. Así, los diagramas especificarán una **anatomía** de la administración, mientras las descripciones especificarán su **fisiología**.

Algunos de los graves problemas que enfrentan las empresas en la actualidad son los siguientes:

- Las estructuras de las empresas son fuertemente dependientes de las personas que cumplen roles conocidos y cuando aquellas personas dejan el cargo, la estructura a menudo debe cambiar.
- El control en la empresa no es algo claro y se toma como si fuera una función de los diferentes grupos de personas interactuando uno con otro. No se tiene la visión que el control tiene que ver con la información en una extensión y complejidad superiores a las capacidades de los gerentes para absorverla e interpretarla.
- No se ha entendido que el computador es una herramienta que posee las capacidades para manejar en forma adecuada el problema del control, siempre que se lo utilice en un esquema correcto.
- No se dispone de un esquema conceptual suficiente, pues si alguna actividad administrativa no se ajusta a ninguna de las divisiones, debe reportarse directamente a la gerencia general.

Por estas razones se requiere de un nuevo modelo organizacional, cuya construcción se ha planteado como objetivo en este trabajo. En el nuevo modelo se usará cuatro nociones clave:

1. Hay una **escala** tanto en tamaño como en complejidad.
2. Hay una **transferencia cruzada**, donde las partes de las cosas actuales se representan de nuevo en sus posiciones relativas.
3. Hay una **trabajabilidad**, por medio de la cual el modelo puede operar tal como el original.
4. El modelo es bueno si es apropiado.

Sigue de estas consideraciones que si se desea pensar acerca del control en la empresa, se debe usar un sistema de control como un modelo. El problema es que los sistemas de control de complejidad suficiente para servir como modelos adecuados de la empresa son tan complicados que no es posible entenderlos a cabalidad, excepto a través de modelos.

La Cibernética compara los modelos de sistemas complejos y analiza los aspectos del control que parecen comunes a todos. Tal sistema debe describir todo lo relacionado con su estructura. El problema consiste ahora en escoger un sistema adecuado para el propósito.

Como una cualidad intrínseca del control, se debe cumplir que la empresa es auto-regulativa en los niveles bajos. Desde la posición del gerente general, hay controladores intrínsecos en el trabajo

que no se hace o que no se hace normalmente. Muchos gerentes generales no sólo no consideran el trabajo de dichos controladores intrínsecos, sino que también lo interfieren.

En el cuerpo humano también se encuentran estos mismos problemas, pero son resueltos adecuadamente. La actividad física es totalmente integral y los conflictos son resueltos en forma armónica.

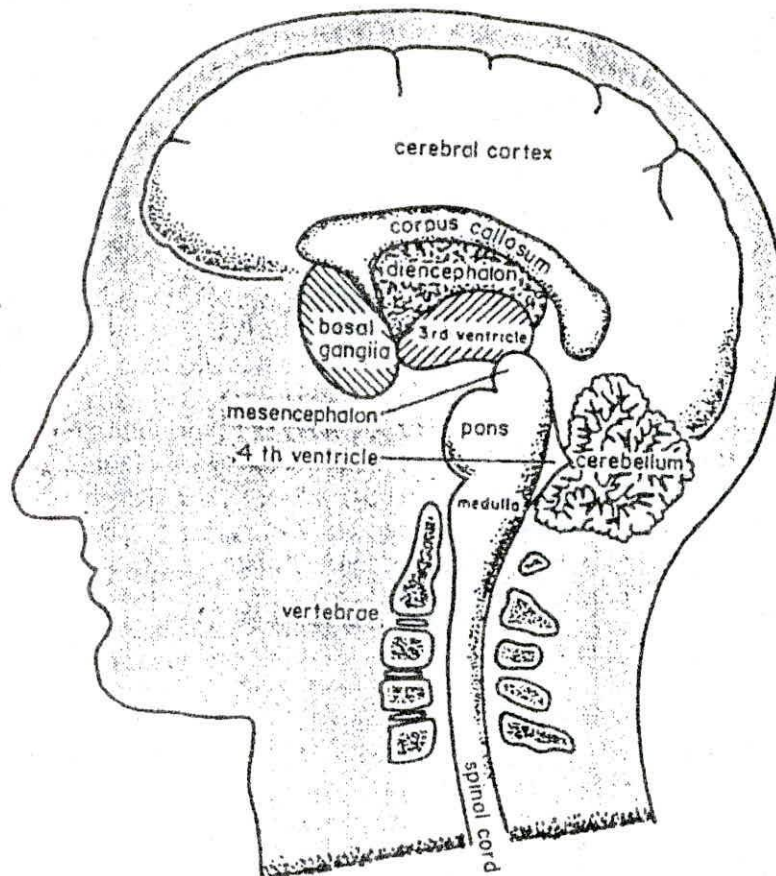
Por motivos como este se escoge como modelo de la empresa el sistema cuerpo humano y su sistema de control, el sistema nervioso. Se verá que el modelo puede manejar en forma adecuada las diferencias en escala, transferencia, trabajabilidad y conveniencia.

Para reafirmar la decisión de usar un modelo neurocibernético, obsérvese que la empresa humana es una condición muy insatisfactoria: sus registros de fallas son enormes en un período de cambio y no se puede estar seguro qué aspectos de la prudencia organizacional son conducentes a viabilidad y cuáles al desastre. El sistema nervioso humano a veces falla también, pero ha resuelto muchos problemas que la empresa aún los tiene. Debe ser exitoso, debido al desarrollo a largo plazo en su estructura y debe haber aprendido en los varios millones de años de evolución.

FISIOLOGIA DEL CONTROL

El cordón espinal, contenido y protegido por la columna vertebral y los nervios que emanan de este cordón recorren todo el cuerpo. El cerebro está en el tope del cordón espinal.

El cordón espinal es el comienzo del sistema nervioso, lo cual significa que es el tipo de estructura nerviosa más antigua. En la evolución llegó primero; el cerebro se desarrolló al final.



Se vio anteriormente que la parte aferente de un sistema de control termina en una placa sensorial, que la placa motor puede originar partes eferentes del sistema y que los dos están conectados por un retículo anastomótico. El tubo que comprende el cordón espinal es exactamente como esto, las placas tienden a ser redondas para formar un tubo. Las entradas llegan por la parte posterior del tubo y las salidas son conducidas hacia el frente.

A lo largo del cordón espinal se puede considerar diversos niveles, desde los cuales salen nervios hacia los órganos del cuerpo. Cada uno de estos niveles se denomina un eje de comandos lateral. De esta forma se tiene un sistema bi-dimensional y en él se basa la habilidad del cuerpo para ejecutar sus tareas autónomamente (trabajando lateralmente) y así integrar la actividad local a un balance orgánico (trabajando verticalmente).

En la parte superior del cordón espinal está el tronco del cerebro, generado por evolución del cordón espinal. También es una especie de tubo y se abre en una serie de hinchazones. Otros órganos son la médula, el bulbo raquídeo, el diencéfalo y el tálamo.

Es importante destacar que todo este aparato no tiene conexión directa con el mundo exterior y que toda la información se origina en los receptores que hacen uso de los nervios como canales de comunicación. Toda la información es procesada a través del cordón espinal y el tronco del cerebro, lo cual puede pensarse como el retículo anastomótico de las partes antiguas del sistema nervioso. Los otros órganos mencionados cumplen funciones especializadas, tal como se verá más adelante y han evolucionado fuera del retículo.

El tronco del cerebro recibe datos provenientes del cordón espinal, pero también recolecta datos relativos a los sentidos altamente especializados (ojos, etc.) a través de sus propios nervios craneanos. Todo este procesamiento y switcheo de datos es el requerido para controlar las acciones involuntarias.

Las acciones voluntarias tienen lugar cuando el tronco del cerebro deja pasar información a la corteza cerebral, donde se toman las decisiones de consciencia; el tronco del cerebro recibe estas instrucciones, las traduce en comandos y las pasa al cordón espinal para la acción.

La médula provee el enlace clave entre el cordón y el cerebro y también contiene los circuitos de switcheo para muchos nervios craneanos.

El bulbo raquídeo lleva las fibras que pertenecen al eje del comando vertical y también fibras cortas para coordinar las dos mitades del cerebelo. En esta parte del retículo hay una notable actividad de filtraje de información. En el mesaencéfalo se monitorea los llamados "reflejos correctos". Este mantiene el equilibrio del cuerpo.

Luego están el diencéfalo, con el tálamo y el ganglio basal, ordenando, switcheando y transmitiendo entre las partes más altas y más bajas del cerebro. Estas partes principales del cerebro son llamadas computadoras especializadas que procesan la información proveniente del eje de comando vertical; cada uno de ellos puede intentar dos tipos de tareas:

1. Inspeccionar la información que está llegando, para determinar si es apropiada de tratar en este nivel. Si es así, sucede lo siguiente:

- Toma una decisión, enviando órdenes por el eje de comando vertical.
- Una versión modificada de la información es pasada hacia arriba.

2. Si el computador especializado no es competente para tomar la acción de comando, hay dos alternativas:

- Dejar pasar la información, sin modificarla.
- Filtrar la información y pasarla hacia arriba.

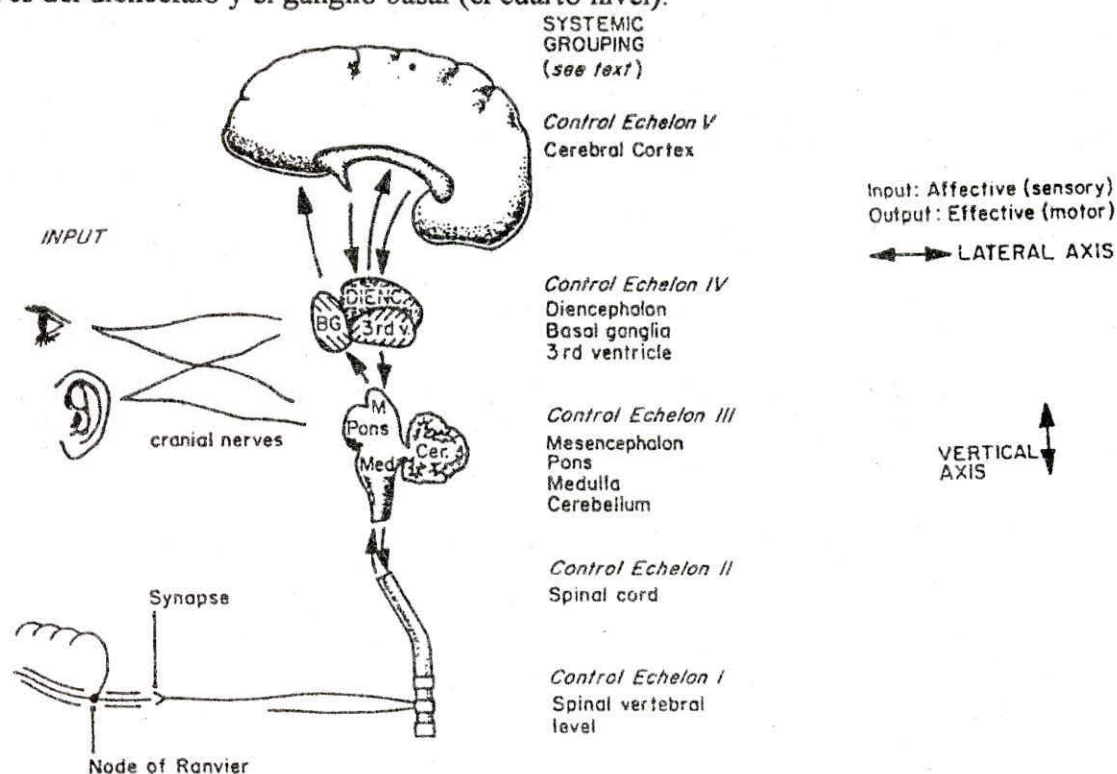
Un filtro es un reductor de variedad. Dos tipos importantes de filtraje en el tronco del cerebro son los mecanismos "excitador" e "inhibidor".

Las partes especializadas del cerebro, como se puede observar, también son un retículo anastomótico.

Yendo ahora al ámbito de la empresa, se puede observar un eje de comando vertical en su interior. Se identifica en términos del hombre y los procedimientos. Si fuera automático, debiera tener un análogo del cordón espinal, coleccionando información y conociendo la acción del eje de comando lateral. La información ascendente enriquecerá el computador central de la empresa, como una parte de la configuración necesaria para integrar las actividades de todas las áreas y funciones. Esta parte de la configuración será el análogo del tronco del cerebro y debe haber un nivel cortical, en el cual el rol de la conciencia es jugado por el consejo de accionistas. Entre estas dos partes debe haber todo tipo de switcheo, ordenamiento y envío, asociados con el diencéfalo y el ganglio basal.

Se puede considerar cinco niveles en la configuración, debidos a las principales diferencias funcionales involucradas. Los cinco sistemas están ligados al eje de comando vertical de la empresa y ellos modelan el sistema nervioso somático del cuerpo. El cordón es el segundo nivel y la médula, el mesencéfalo, el cerebelo y el bulbo raquídeo conforman el tercero. El cuarto nivel es el diencéfalo, con el tálamo y el ganglio basal.

El computador especializado que enlaza el tercer nivel (tronco del cerebro) al quinto nivel (corteza cerebral) es lo que la administración llama la función de staff. Todos los nervios sensoriales reportan al tálamo; cada elemento que obtiene la corteza cerebral es ordenado y switcheado a través del diencéfalo y el ganglio basal (el cuarto nivel).



Tal como la corteza cerebral no está en contacto directo con los eventos periféricos, sino que recibe los datos de los niveles subordinados, así también el tope de la administración se debe asumir aislada de los eventos corrientes.

De lo descrito del funcionamiento del sistema nervioso, se deduce que el staff en la empresa debiera estar abiertamente implicado en el proceso de toma de decisiones, con las responsabilidades que le corresponda.

En el primer nivel se captura los datos y se los procesa inicialmente. El cuerpo, tal como la empresa, tiene receptores que registran información. Parte de la información del entorno es capturada por los exteroceptores. Los interreceptores capturan datos relativos a los estados internos del organismo. Finalmente están los prioreceptores, que sirven al sentido kinaestético, que tiene relación con la posición corporal.

El eje lateral, que es el nivel más bajo del sistema, colecciona y disemina información. Esta es pasada a través de los tres niveles principales del eje de comando central y, finalmente, en el quinto nivel, alcanza la corteza cerebral. En el curso de este proceso se toma una vasta cantidad de acción de control.

La corteza necesita la salida requerida por una sensación y también necesita facilidades para salidas que iniciarán una acción. Aquí se distingue lo sensorial y la corteza motor; cibernéticamente pueden ser vistas como las últimas placas de los sistemas de entrada y de salida. La corteza tiene relación con el intelecto; es el asiento de la conciencia.

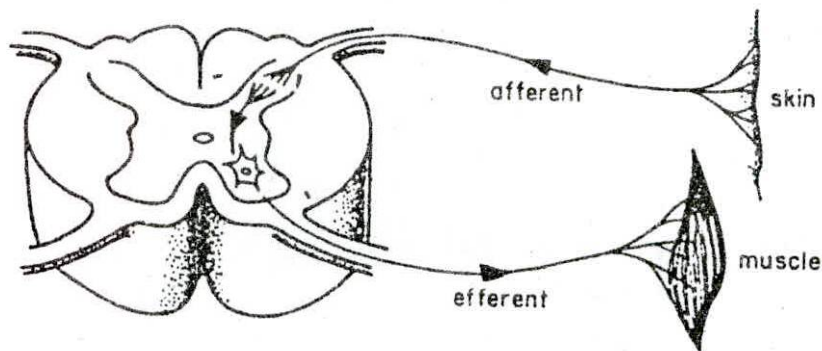
AUTONOMIA

Desde el punto de vista del organismo, sea el cuerpo humano o la empresa, la función autonómica es esencialmente mantener un ambiente interno estable, es decir, ser homeostático. La producción bien ordenada no debe ser excesiva, en términos tanto de hombres como de máquinas; los costos y la calidad deben mantenerse dentro de los límites fisiológicos, lo que significa que pueden variar dentro de un rango estrecho para el buen estado del organismo completo; los stocks de los materiales interprocesos deben mantenerse lo suficientemente bajos para evitar tiempo perdido.

Todas estas cosas pueden salirse del control, ya sea por cambios en el ambiente externo como por razones internas. Cualquiera sea la causa del desbalance, el control autonómico debe corregirlo. Los receptores detectan el cambio, traduciéndolo en impulsos aferentes, los que van al centro de control y computados, haciendo el ajuste a través de la parte motor del sistema. Esto es lo que se denomina **reflejo autonómico**.

En la empresa esto es una función de la administración media y en el cuerpo humano reside en la sección media del cordón espinal, llamado sistema nervioso simpático.

La parte básica del sistema nervioso simpático funciona enviando un mensaje desde algún receptor hacia el cordón espinal, desde el cual emana un mensaje a un músculo. Esto se denomina arco reflejo y puede verse en la siguiente figura:



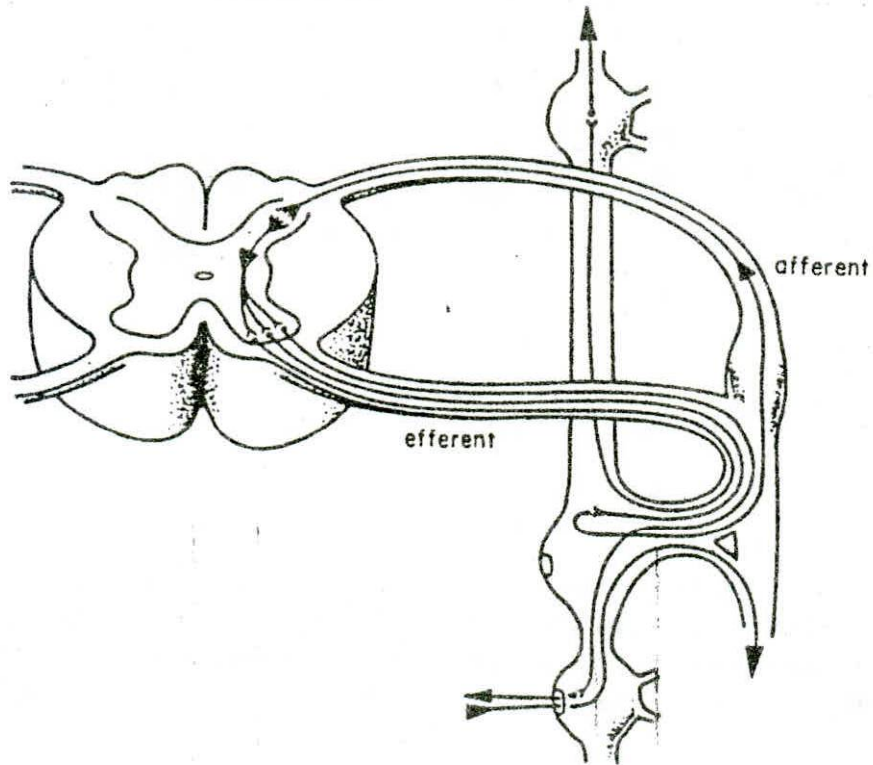
Similarmente y como ejemplo, se tiene que un administrador departamental ejerce el control de los costos; si un receptor detecta una variación, es decir, una discrepancia entre los costos estándar que se han convenido y los costos reales incurridos, un impulso aferente pasa al administrador, el que toma una decisión y envía un mensaje a los efectores en la tienda para corregir la perturbación.

Como se ha visto, debe haber partes aferentes y partes eferentes en un sistema de control, entradas y salidas cuyos mensajes son traducidos por receptores y efectores, respectivamente. Entre ellos debe haber un dispositivo de switcheo de alguna clase. Más aún, tal como se vio anteriormente, tal circuito debe ser descrito mejor como una retroalimentación controlada negativamente, más que un emisor de instrucciones.

La descripción dada del arco reflejo es una primera aproximación. En la realidad, ningún sistema de control en un sistema viable es bien localizado y autosuficiente como se explicó, sino que toda acción correctiva que se tome en un lugar debe considerar los efectos sobre otras actividades próximas.

Los administradores en los diferentes niveles del eje de comando central son influenciados por las decisiones locales o al menos necesitan saber de ellas, pues ellos conocen la actividad planificada desde el nivel más alto en el sistema.

Las principales vías de comunicación hacia arriba y hacia abajo del eje de comando central se usan para interrelacionar las actividades de los diferentes departamentos y funciones dentro del plan total. Esta es una actividad suficientemente compleja, tanto para las motoneuronas del cordón espinal como para los diferentes controladores de la empresa. Esto se debe a que en el arco reflejo del sistema autónómico la administración local es descentralizada, pero debe estar coordinada con las otras actividades similares a lo largo del cordón.



Observando este nuevo diagrama del arco reflejo, se tiene que tanto la neurona aferente, que transmite la información de entrada acerca del desalineamiento, y las neuronas motoras, que dan efecto al cambio en el plan, están ubicadas fuera del eje central.

La neurona aferente envía información hacia el eje central sólo después que ha notado que hay información significativamente suficiente acerca del proceso que se está sensando y que excede su umbral. Similarmente, la motoneurona central no actúa y cambia de plan sino hasta que su umbral también sea excedido.

Los ganglios periféricos interactúan a través del tronco simpático, que está fuera del eje de comando central.

Por otra parte, desde el cráneo surge el llamado sistema nervioso parasimpático, que cumple una función marcadamente diferente de la del sistema simpático y sus nervios se distribuyen en todos los órganos del cuerpo, llevando información acerca del funcionamiento de éstos directamente al tronco del cerebro.

El sistema descrito tiene por objetivo su homeostasis. El sistema simpático es una función de la administración media y el sistema parasimpático es una función de la administración superior. Así, las actividades de la parte autonómica son controladas por dos sistemas diferentes.

Los próximos dos niveles de control están relacionados con interpretar las instrucciones de alto nivel en patrones de actividad más detallada. Ellos también son responsables para iniciar las señales de retroalimentación que capacitan al controlador superior para hacer planes predictivos. Al hacerlo, los datos de retroalimentación son reducidos a los controladores de niveles bajo e intermedio, tal que sólo la información significativa sobre la exactitud de la predicción es retroalimentada al controlador principal. Esta actividad no impide que en un momento dado haya intervención de control desde un nivel superior.

ADMINISTRACION AUTONOMICA

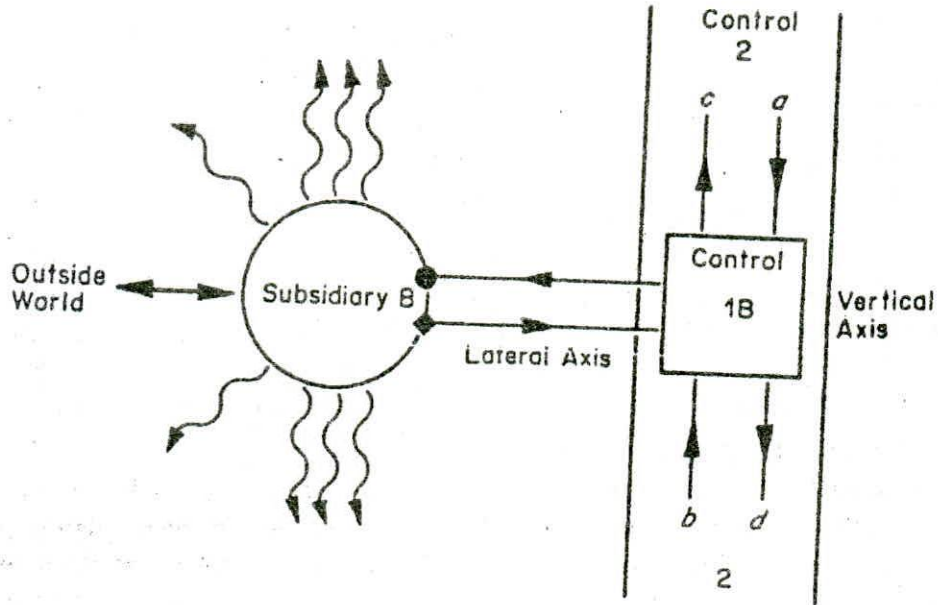
Se ha estudiado la fisiología de un sistema de control en términos de un modelo del sistema nervioso humano, analizando su estructura con el fin de tener una base acerca de cómo trabajar.

Si se considera una gran empresa, se encontrará que la organización completa está fragmentada, en forma similar que los órganos del cuerpo humano, pero, a diferencia de éste, lo más probable es que la administración también está fragmentada, de tal manera que en lugar de la corriente ascendente de información que pasa a través de una jerarquía de sistemas computacionales, se encontrará información que sube, relativa a cada una de las divisiones funcionales por separado, con muchas dificultades de comunicación entre ellas.

El modelo neurocibernético entrega una solución a este problema al considerarse como un plano por el cual los computadores se deben interconectar para obtener un sistema de jerarquía quintuple con el fin de proveer a la empresa de una herramienta para la construcción de su propio sistema de control. Este se propone a continuación.

Considérese una empresa con un eje de comando vertical, consistente de cinco sistemas de computación arreglados jerárquicamente. Considere una subsidiaria de esta empresa, que puede ser una compañía completa o un departamento, en forma análoga a un órgano del cuerpo humano.

Este es conectado al eje de comando vertical, en el nivel del Sistema Uno, por un eje de comando lateral. Esto se puede representar mediante la siguiente figura:



La subsidiaria (llamada B) ejecuta la acción, la que es señalada en la figura por las flechas ondeadas. Lo que está realizando esta subsidiaria debe ser comunicado por el Sistema Uno al Sistema Dos, el cual es el análogo del cordón espinal. En la figura se muestra tal "vértebra" del Sistema Dos y es el centro de comandos del Sistema Uno.

Debido a que la subsidiaria es parte de la corporación completa, se sensa sus actividades por interoreceptores, los que son representados por el diamante negro en la figura. La información es enviada a lo largo del acceso lateral a la intersección de los Sistemas Uno y Dos y la respuesta es devuelta a la subsidiaria.

Las instrucciones para esta subsidiaria llegan a la vértebra apropiada del Sistema Dos, llamada Sistema Uno, desde los sistemas de nivel más alto en la línea marcada a y las instrucciones

diseñadas para otras subsidiarias pasa hacia abajo por la línea marcada d.

Los datos acerca de otras subsidiarias necesarios por niveles más altos de control suben a través de la línea de transmisión marcada b, donde se le une nueva información de este centro y procede a subirla por la línea c.

Se puede idear arreglos que informen al Sistema Dos de las subsidiarias, a través de sus propios controles de actividad del Sistema Uno. También se puede registrar, mediante el uso de computadores interconectados y métodos apropiados de ingreso de datos, las actividades de la subsidiaria en todos los niveles del sistema de control corporativo.

La vértebra del Sistema Dos, que es el controlador del Sistema Uno de la subsidiaria B, tiene un conjunto de instrucciones básicas transmitidas por la línea a, junto con las modificaciones que se producen momento a momento e instrucciones especiales, que deben ser comparadas con las entradas traducidas por la subsidiaria. Aquí se produce un punto crítico, que se describe a continuación.

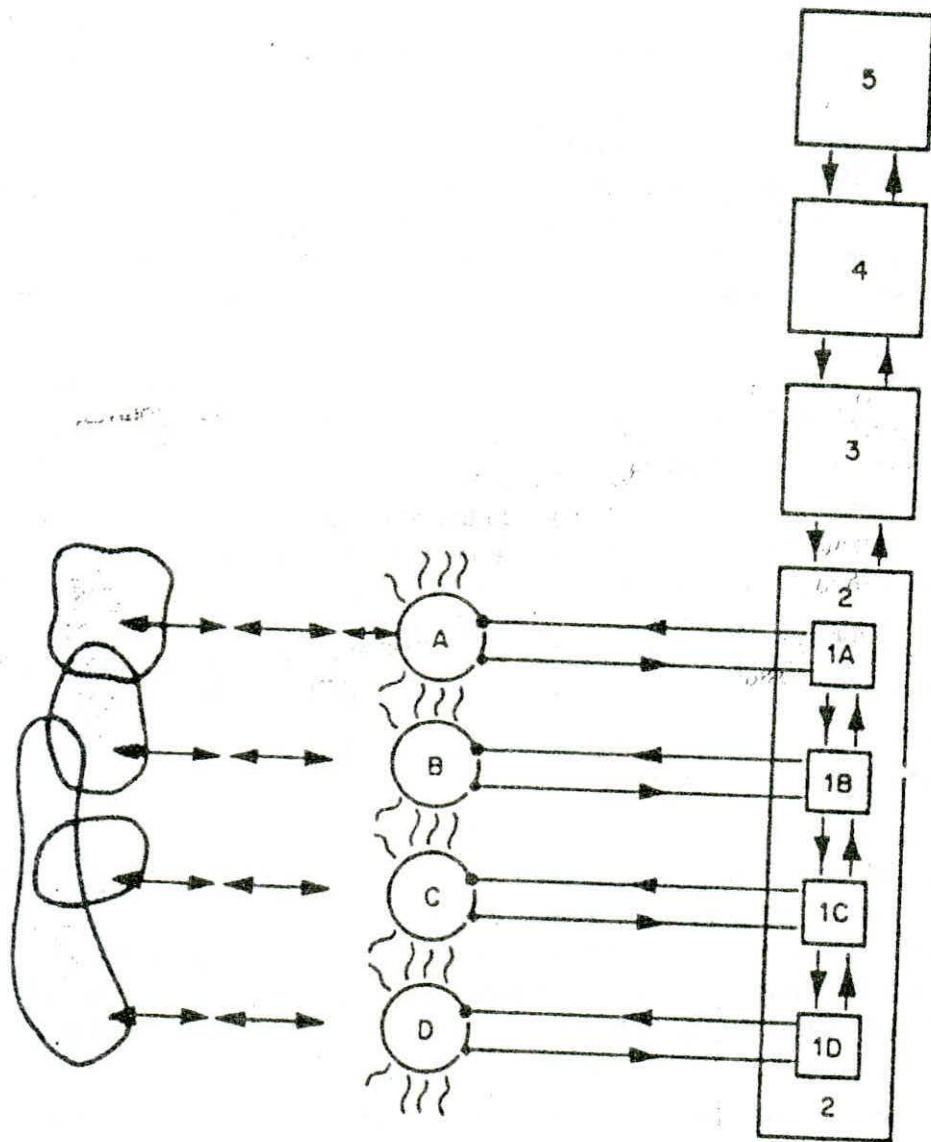
El Sistema Dos tiene un plan inicial para el ensamblaje de las subsidiarias y esta vértebra particular (1B) tiene un plan inicial para la subsidiaria B. Así, el controlador 1B es capaz de comparar el rendimiento actual con este plan y también modificar el plan al recibir nuevas instrucciones, a medida que llegan.

Hay muchas razones por las cuales las actividades de la subsidiaria pueden fallar de acuerdo con la intención original y por lo tanto se deben hacer ajustes. La tarea principal del Sistema Uno es computar tales ajustes rápidamente y alimentar a la subsidiaria. Es importante notar que ésta es una actividad refleja. Gracias a las técnicas predictivas estadísticas es posible para el controlador 1B anticiparse a los desvíos del plan y comenzar a modificar sus instrucciones en avance.

Continuando con la descripción de la labor del eje lateral, para que ésta se cumpla deben satisfacerse las siguientes condiciones:

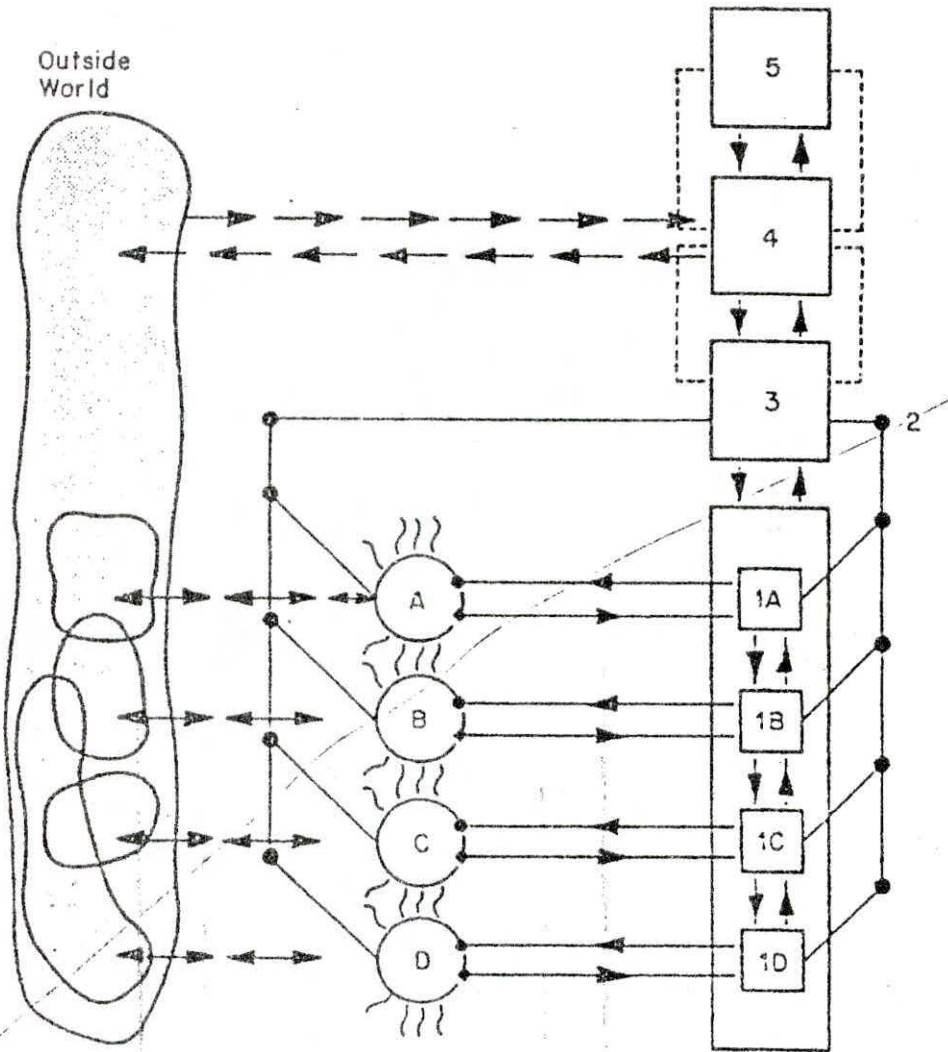
- Debe haber un plan inicial.
- Debe haber una modificación constante del plan en el eje de comando central, con el fin de satisfacer las necesidades del organismo como un todo.
- Debe haber un reconocimiento inmediato del estado actual de la situación; de lo contrario, se puede introducir retrasos en el ciclo reflejo, pudiendo llegar a una oscilación descontrolada.
- Debe haber una forma de comandar la susidiaria, con el fin de modificar sus planes para superar cualquier dificultad que se presente.

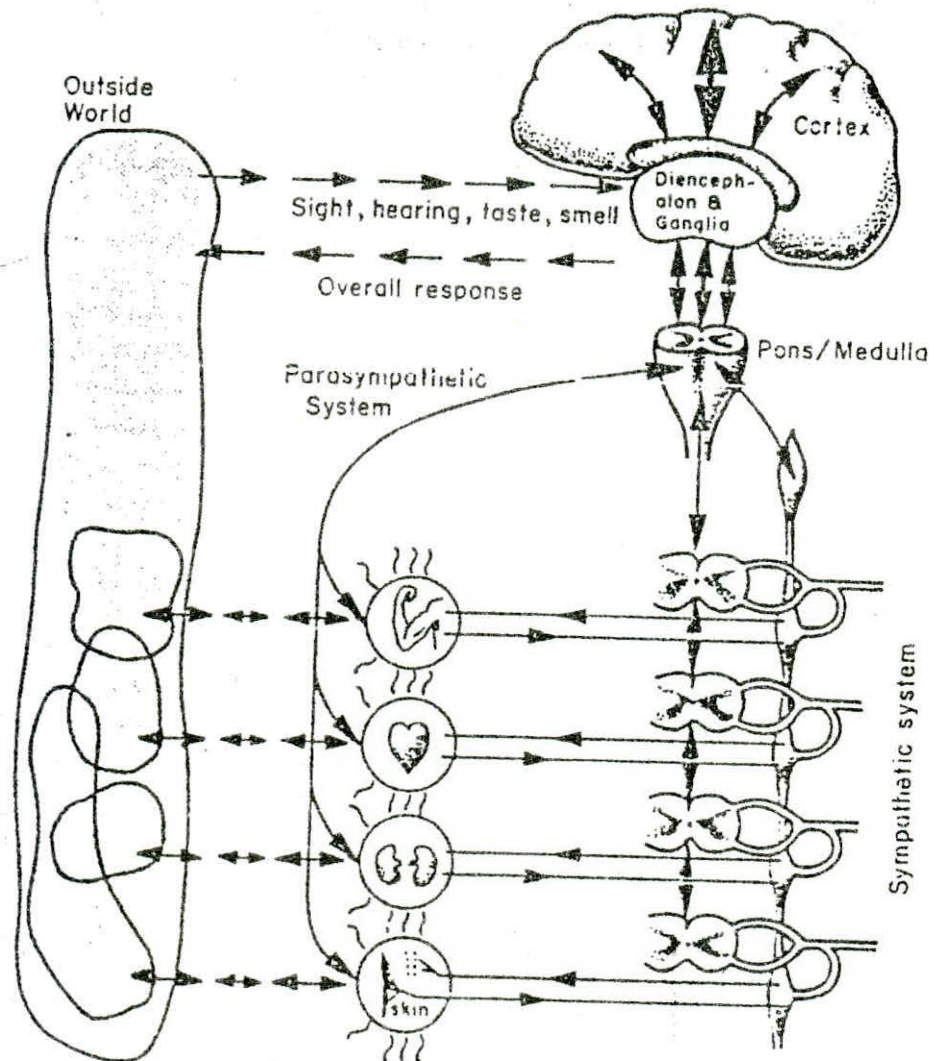
El Sistema Dos puede solucionar algunos de estos problemas al proveer interacción local entre los Sistemas Uno de todas las subsidiarias, como se observa en la siguiente figura.



En la figura se tiene el análogo del cordón espinal, donde están situados todos los Sistemas Uno y se observa también cómo cada uno de ellos se pone en comunicación con sus similares, a fin de actuar concertadamente en la línea descendente a que está dando las órdenes.

Se puede producir problemas cuando el proceso de control (Sistemas Uno sumergidos en el Sistema Dos) se concibe como una actividad dinámica, es decir, se considera un sistema autónomo. Para analizar esto, obsérvese la siguiente figura, donde se agrega el sistema autónomo neurofisiológico.





El lado derecho de la figura que describe el sistema organizacional es el análogo del tronco simpático, que enlaza los nodos vertebrales del Sistema Dos. El lado izquierdo es el análogo del conjunto de nervios parasimpáticos. Estos conforman el Sistema Tres, que controla la estabilidad del ambiente interno de la empresa y hace esto utilizando retroalimentación.

Cada órgano del cuerpo, que puede ser llamado la subsidiaria de la empresa, puede actuar aisladamente en su eje lateral si no fuera por el arreglo de cada controlador de los órganos en un conjunto cohesivo de tales controladores, que se llama Sistema Dos. Pero el Sistema Dos podría actuar sin objetivo si no fuera monitoreado en un nivel más alto por el Sistema Tres.

Considérese la primera producción en la empresa y supóngase que cada una de las subsidiarias tiene un rol que jugar en la manufactura del conjunto principal de productos de la empresa. Por ejemplo, la subsidiaria B manufactura ciertas salidas, algunas de las cuales pueden venderse directamente, como se muestra en la figura mencionada, pero otra cantidad se pasa a la subsidiaria C para mayor proceso. Después de eso, algo de la salida de C será pasado a la subsidiaria D y así sucesivamente.

Supóngase que algunas de las tareas del programa de producción de la subsidiaria C son mal realizadas. Su controlador, 1C dentro de 2, intentará ajustar la actividad al plan acordado, pero será imposible hacerlo localmente, en el sentido que el acuerdo para obtener suministros desde la subsidiaria B puede no variarse y el acuerdo para hacer las entregas a la subsidiaria D puede no cumplirse. El controlador 1C dentro de 2 debe informar a los controladores 1B y 1D y así todos tratarán de cambiar sus planes para llegar a un nuevo acuerdo.

El sistema autonómico (simpático) en el lado derecho de la figura monitorea toda la actividad descrita anteriormente. Usa un lenguaje de orden más alto que el del Sistema Dos, debido a que debe discernir sobre el comportamiento de éste. Como su tarea es estabilizar el ambiente de producción de la empresa, debe suministrar realimentación en los diversos niveles, que tenderán a disminuir las oscilaciones causadas por los ajustes de la replanificación.

Aún así, se puede llegar a tener una actividad frenética y en este caso los centros de alto control en la empresa deben entrar en acción para manejar la crisis, lo cual será sentido en todas las subsidiarias. El Sistema Uno responderá yendo directamente a los computadores del Sistema Dos, los que harán una racionalización local y comunicarán a los centros de control del Sistema Tres a través del sistema central. Pero la misma información subirá por el tronco simpático y llegará al centro de control por otra ruta.

La retroalimentación estimuladora funciona de tal forma que al lado derecho de la figura hay un ciclo de actividad excitatoria que trata de satisfacer lo establecido por el Sistema Tres. Pero supóngase que esto conduce a forzar mucho a las subsidiarias. Por ejemplo, los índices de productividad miden que la razón de producción puede subir sobre los límites superiores de control en vigencia y el nivel de sobretiempo trabajado también puede subir peligrosamente. Entonces los procedimientos de control pueden llegar a ser inestables, debido a que hay mucha actividad.

Dichas señales serán registradas en la red autonómica (parasimpático), mostrado a la izquierda de la figura. Estos datos también subirán a los centros de control del Sistema Tres. La respuesta debe ser inhibir la actividad, mediante un ciclo que va por el lado izquierdo de la figura.

Se ha mostrado cómo el sistema autonómico balancea los cuadros excitatorios e inhibitorios, a fin de producir una estabilidad interna completa. Esto se reporta hacia arriba, a través del Sistema Cuatro, para llegar al Sistema Cinco, donde fue formulada la política.

El consejo de accionistas define las políticas en base a la información que llega desde los exteroceptores y los interoceptores a través del Sistema Cuatro, que cumple la función de staff. Este último clasifica la decisión del nivel superior y envía comandos hacia abajo a los ejes vertical y lateral.

El Sistema Tres, el autonómico, tratará de velar de que el plan sea implementado, reportando también las modificaciones que se le hagan.

En la empresa, habitualmente los gerentes de producción y de ventas tienen intereses diferentes. La principal tarea de control será poner esos intereses en acuerdo y dentro de esto, se tiene que el Sistema Tres es el centro de asignación de recursos, utilizando técnicas provistas por disciplinas tales como la Programación Dinámica y la Programación Lineal.

EL SWITCH MAS GRANDE

Se ha llegado a un punto donde la estabilidad interna del organismo está asegurada por los tres sistemas de más bajo nivel, conocido como control autonómico. En esta última sección se trata con la operación global de los cinco niveles del cuerpo corporativo. Esto conduce a analizar al Sistema Cuatro, que es el mecanismo de enlace entre el control volicional y el autonómico; es el switch más grande de la organización.

Un switch es un dispositivo que transfiere señales desde una parte de un sistema a otro. En este trabajo se ha tratado con varios switches, tanto en el aspecto fisiológico como administrativo y se vio que no son mecanismos simples, sino que son algedonodos, razón por la cual puede excitar o inhibir la señal que deja pasar. La actividad de inhibición es muy importante, debido a que evita el problema de sobrecarga en los canales de comunicación, además de ser útil en el manejo de la variedad del sistema.

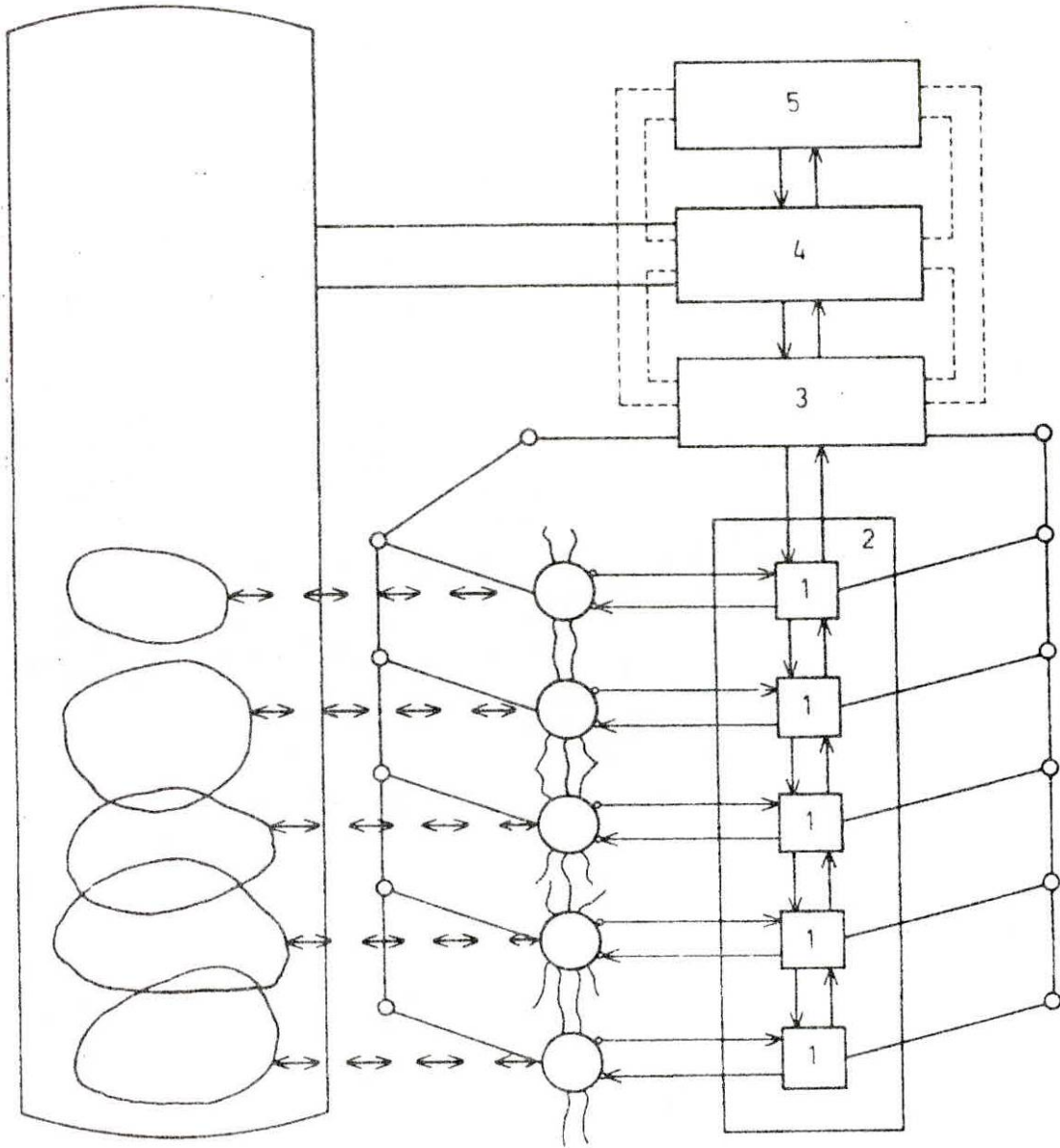
Tal como se describió, en el sistema nervioso hay un mecanismo dual, en el cual se balancea la excitación con la inhibición, más que un solo dispositivo de encendido. Lo mismo es cierto en administración, donde la mayoría de las señales de entrada pueden ser atenuadas o amplificadas por algún switch, de tal forma que éste no debe ser visto como un mecanismo on/off, sino como un algedónodo, ya que la decisión que tome está gobernada por un conjunto complejo de entradas excitatorias e inhibitorias, con un valor umbral variable.

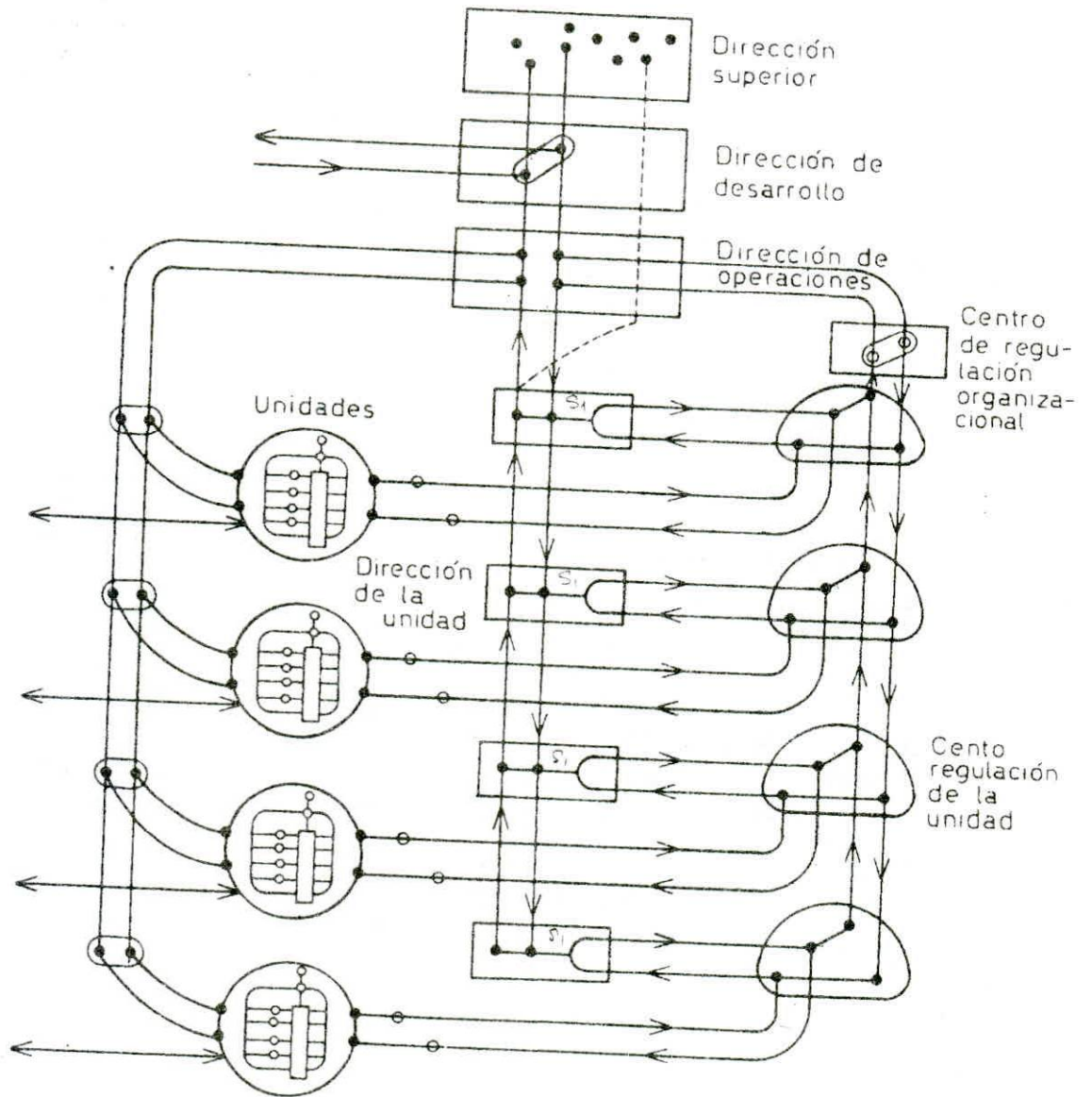
Algunos algedónodos que se ha estudiado son los órganos terminales receptores, los ganglios con sus umbrales y el gran centro reflejo interior del cordón espinal, las neuronas y la corteza cerebral.

El switch más grande de todos es el Sistema Cuatro, pero también es el más complejo, en el sentido de que es modulador probabilístico, debido a que los mensajes que pasan a través de él deben ser "subidos de tono" o "bajados de tono", lo cual significa subir o bajar la probabilidad condicional con que el switch transmite el mensaje. Este es el mecanismo excitación-inhibición que utiliza dicho switch.

En este punto, es importante analizar algunos aspectos del Sistema Tres, el cual está totalmente alerta en los niveles que están debajo de él, pero hacia arriba, sus controladores son esencialmente inhibidores, pues, de lo contrario, los niveles de control más altos podrían ser inundados con información de poca o ninguna importancia estratégica.

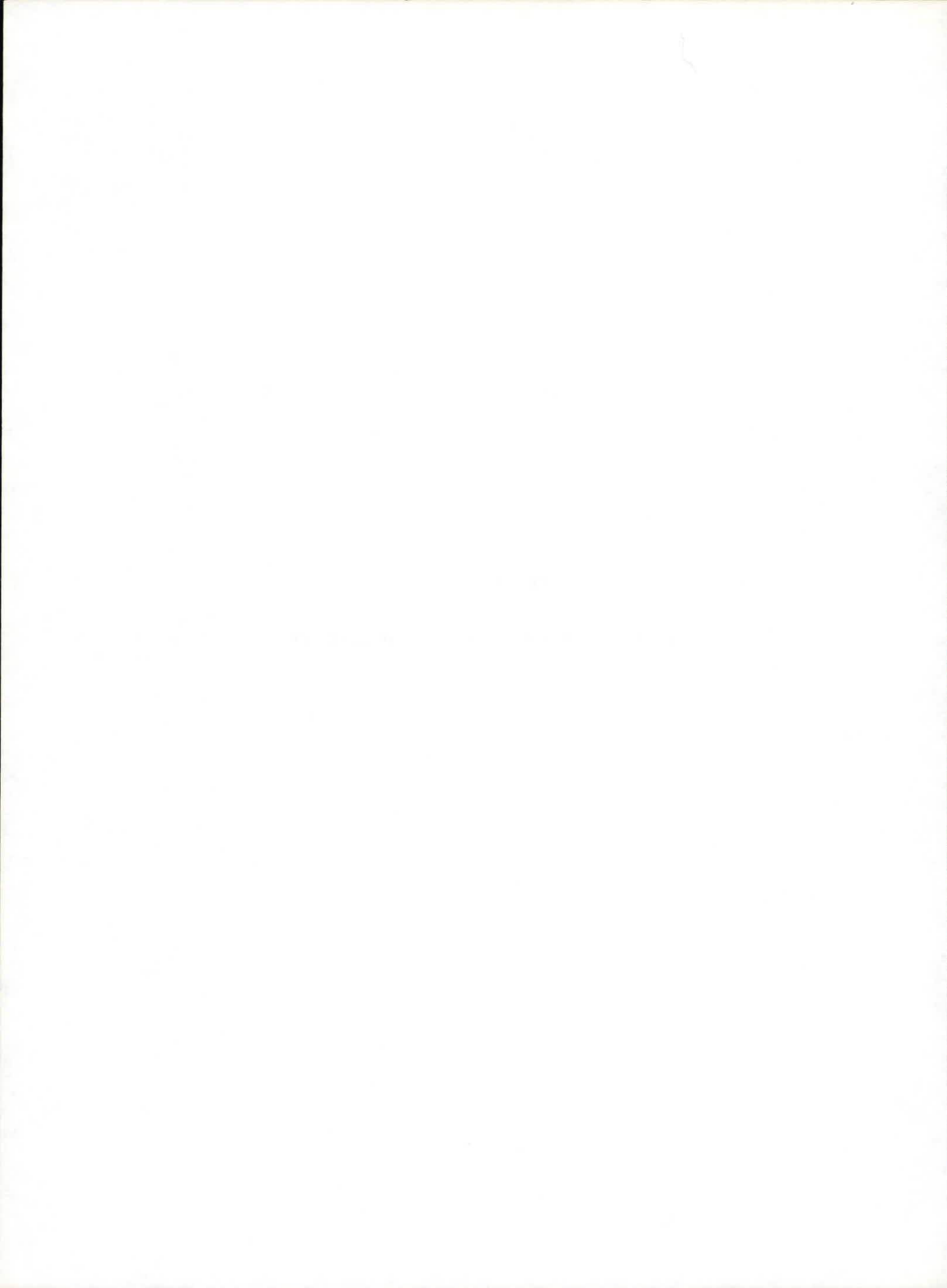
Así, el Sistema Cuatro cumple un papel fundamental. Enlaza el nivel estratégico a su estructura corporativa y constituye una colección ramificada de algedónodos para switchear hacia abajo todos los requerimientos volicionales del cerebro. Igualmente switchea hacia arriba toda la información requerida por la corteza cerebral para hacer funcionar el cuerpo, incluyendo representaciones apropiadas (debidamente filtradas) de la función autonómica. Captura todos los datos relativos al ambiente a través de los sentidos, los filtra y redistribuye información relevante tanto hacia abajo como hacia arriba, utilizando a todos los otros escalones de control. Finalmente, opera lo que se ha llamado sistema algedónico mismo, mecanismos excitadores e inhibidores, los cuales tienen nervios especiales aferentes y eferentes.





SECCION C

LOS SISTEMAS INFORMATICOS EN LA ORGANIZACION



I. EL ROL DE LOS SISTEMAS INFORMÁTICOS EN LA EMPRESA.

El análisis de sistemas consiste en identificar y especificar el conjunto de subfunciones o subprocesos que han de conformar el sistema de información o sistema de procesamiento de datos a construir. Para ello es preciso identificar la estructura de la función objeto de desarrollo informático. Esto significa que debe ser descompuesta en subfunciones, destacando preferentemente las interfaces que las unen, por cuanto son éstas a través de las cuales fluyen los datos o información inherente al sistema.

Ello es así por cuanto, una vez establecida la trama estructural de la función, ésta debe someterse a estudio con el propósito de definir los cambios estructurales necesarios, ya sea incorporando, eliminando, agregando o separando subfunciones, para así establecer cuáles de ellas serán automatizadas y cuáles serán manuales, todo ello con el propósito de reinsertarla nuevamente en la organización una vez concluida su construcción y su implementación, ahora redefinida estructuralmente y con nuevos niveles de soporte para el tratamiento de datos y de generación de información.

En consecuencia, se puede deducir que uno de los propósitos que se persigue con el sometimiento de una función o proceso a tratamiento informático es aumentar su efectividad como componente orgánico, por lo cual necesariamente debe entenderse como componente estructural, ligado indisolublemente a otras subfunciones.

Por ser componentes estructurales, los sistemas de información son justamente eso, sistemas, pero sistemas que además de insertarse efectivamente en la red de procesos, se muevan dinámicamente al mismo ritmo de funcionamiento del resto de los componentes de la estructura.

Por lo tanto, este proceso de enfrentar la función empresarial, someterla a tratamiento informático y reinsertarla nuevamente en la estructura de la cual se extrae, sólo será efectivo en la medida en que se tenga absoluta claridad respecto del rol estructural de los sistemas a desarrollar, es decir, de su rol específico y parcial, de sus relaciones con los otros sistemas y con los otros componentes estructurales.

Ello exige, como fundamento previo, un conocimiento acabado de la estructura organizacional en la cual aquellos sistemas desarrollan su quehacer, misión que cumplen ordenados en una conformación esencialmente jerárquica.

El análisis de las estructuras organizacionales, a objeto de identificar el rol fisiológico de la información y el rol estructural de los sistemas de información, debe permitir, además, perfilar una perspectiva real de la verdadera dimensión del problema informático en la organización, la cual, a su vez, provee los elementos necesarios para la formulación de las políticas y los planes informáticos, base esencial para enfrentar y resolver con efectividad el desarrollo de sistemas.

1. La Dimensión Estructural de los Procesos.

El nivel de procesos operacionales está determinado por una red de procesos, en donde cada uno de ellos lleva a cabo una determinada transformación, la cual representa el rol que a cada uno de dichos procesos le corresponde dentro de la Empresa.

Pero estas transformaciones no pueden ser cualquier transformación. Ellas deben responder a los subobjetivos asignados al proceso. Para asegurar que esto ocurra, es preciso que en

éstos se manifiesten ciertas instancias de control o regulación, mediante las cuales la transformación puede asegurar el logro de aquellos subobjetivos. Aquel componente que al interior del proceso asume la responsabilidad por el logro de los objetivos se puede denominar *conducción*. Para cumplir con las tareas de regulación y control, éste opera a través de *variables de control*, las cuales son aquellas mediante las cuales es posible verificar tanto el grado de ajuste de la conducta real de la transformación a ciertos objetivos esperados, como la capacidad operacional del proceso. Estas dos dimensiones de las variables de control permiten distinguir dos categorías: las variables de control operacional y las variables esenciales.

Las *variables de control operacionales* son aquellas que permiten verificar el ajuste de los resultados de la transformación a los subobjetivos asignados al proceso. Dichos subobjetivos llevan implícitos, además, los valores esperados de las variables de control operacional. Por lo tanto, para materializar la regulación, es preciso que la transformación provea de los valores correspondientes a su conducta real, de tal modo que sea posible determinar los alcances de los desajustes y, en función de ellos, aplicar las acciones correctivas necesarias y asegurar así el logro de los objetivos asignados en eficiencia y eficacia.

Además, los procesos deben mantener el equilibrio de su medio interno, que es aquel que garantiza la mantención de la capacidad potencial de realizar el trabajo que orgánicamente le corresponde. Para ello, deben conservarse en sus valores normales las diferentes variables fisiológicas que lo determinan, y que también forman parte de sus variables de estado. Estas variables, considerando lo sugerido por Ashby, se pueden denominar *variables esenciales*, y los sistemas en los cuales esto ocurre se denominan homeostáticos, en donde la regulación y el control son los mecanismos que le permiten alcanzar tal categoría.

Estas *variables de control* tienen su origen en las variables de estado que definen el comportamiento del proceso y son, en consecuencia, los vehículos a través de los cuales se definen el comportamiento del proceso y son, en consecuencia, los vehículos a través de los cuales se materializa la retroalimentación, entendida como el flujo de información que permite corregir las desviaciones que se pueden producir entre la conducta esperada y la conducta real que manifiesta la transformación.

En consecuencia, para comprender el funcionamiento de una transformación, es preciso observar cómo son absorbidos los objetivos al interior del proceso en las instancias de regulación y control (el nodo de conducción), cuál es la relación entre esta instancia de absorción de objetivos y la transformación, y cuál es el rol estructural que en esta relación cumplen las variables de control.

2. El Rol de las Transformaciones.

Toda transformación, al interior del proceso, representa una conversión del tipo insumo-producto. Ésta puede ser de materia prima en producto terminado, o bien de un documento simple en un producto elaborado, como ocurre, por ejemplo, en la transformación de la orden de venta en factura. Más específicamente, dicha transformación representa la materialización de la misión que la Empresa le encomienda al proceso.

Una característica fundamental de estas transformaciones, es que ellas se insertan en la dinámica orgánica a través de flujos eminentemente materiales, siendo esto válido tanto para

procesos productivos como para aquellos que comunmente se conocen como *administrativos*, y cuya misión se concreta en la conversión de ciertos documentos en otros documentos. No les compete, por ende, la transformación de información.

En cuanto a lo que información se refiere, son sólo emisores y receptores, ya que los flujos de información circulan por el nodo de conducción, nodo con el cual orgánicamente se relacionan al interior del proceso, y que es aquel que, en cuanto instancia de regulación o control, tiene como misión orgánica acoger los objetivos asignados, activar, a partir de ellos, la transformación, y ajustar su comportamiento frente a eventuales desviaciones respecto de los objetivos esperados.

3. El Rol de la Conducción.

A cualquier nivel de la estructura jerárquica, al nodo de conducción le corresponde materializar las tareas de gestión, es decir, la finalización y todos los mecanismos de control o regulación que permiten el funcionamiento y la constante adaptación de los procesos. Es éste el nodo que recibe los subobjetivos y políticas de la empresa y los transmite a las instancias de transformación. Dispone para el cumplimiento de su misión de dos tipos de variables de acción o de gestión: las variables de finalización y las variables de ajuste.

Las *variables de finalización* son aquellas mediante las cuales se activa el nodo o módulo de transformación a través de instrucciones que le transmite, y que se derivan de aquellas que recibe, a nivel de subobjetivos asignados, de los niveles superiores. En los niveles inferiores de la estructura de los procesos operacionales, un trabajo menor (semanal o diario) derivado de un programa mayor (anual o mensual) es un ejemplo de este tipo.

Las *variables de ajuste*, por su parte, son aquellas que permiten, tras un proceso de toma de decisiones, alimentado con la información que da cuenta de lo sucedido durante la transformación insumo-producto, corregir eventuales desviaciones entre lo real y lo esperado. Sus alcances dependerán del rango de autonomía decisional de que dispone la conducción dentro del marco definido por el nodo de gestión global, en cuanto instancia de control, y más específicamente, por las variables de finalización. La asignación de personal a las tareas y las reasignaciones de acuerdo al desarrollo de ellas, son ejemplos de estas variables.

En este contexto, el problema de la gestión al interior de los procesos operacionales, a cualquier nivel de desagregación, se traduce, en primer término, en la activación por parte de la conducción, del nodo de transformación, y en el posterior ajuste de su quehacer, de tal forma que se adscriba a lo esperado, verificando a la par la capacidad operativa del proceso. Para ello, como se señaló, se definen las variables de control.

4. El Rol de las Variables de Control.

Las variables de control son instrumentos de regulación o control y, tal como se indicó, de dos tipos: de control operacional y esenciales.

La tarea de regular la conducta del nodo de transformación, inserta en la acción de controlar que ejerce el nivel jerárquico inmediatamente superior, sólo será posible en la medida en que se cuente con información relativa al comportamiento de la transformación, de manera de

poder cotejar lo real con lo esperado, tanto en relación a los resultados de la actividad del proceso, como en cuanto al estado de su capacidad operacional para obtenerlos.

Las variables de control están adscritas a la dupla conducción-transformación. Dentro del esquema sugerido, están conectadas al módulo de transformación, y más específicamente, se derivan de las variables de estado que definen el comportamiento de la transformación. Se constituyen, por lo tanto, en un instrumento de gestión para la conducción.

Las variables de control operacional se agrupan en torno a ciertos *vectores de control operacional* (VCO), cada uno de los cuales está compuesto por un conjunto de variables relacionadas de alguna manera.

Lo mismo ocurre con las variables esenciales. Éstas también se agrupan en torno a vectores, en este caso denominados *vectores esenciales* (VE).

5. Vectores de Control, Bases de Datos, y Sistemas de Información.

Los vectores esenciales y los vectores de control operacional, en cuanto instrumentos de regulación o control, tienen como misión estructural relacionar los nodos de conducción y transformación. Dicha relación se manifiesta en la entrega de la información que requiere la conducción para regular las transformaciones, apoyando de ese modo el logro de los objetivos en eficiencia y eficacia.

Para cumplir su rol, estos vectores requieren del uso, y, por ende, de la implantación previa de instrumentos de medición, registro, transmisión, y tratamiento de datos, que permitan dar cuenta del carácter que ha tenido el logro de los objetivos.

Se señaló que los valores esperados en relación a la conducta de la transformación se encuentran inscritos en los objetivos asignados. Pero la tarea reguladora requiere también de los valores reales, y éstos deben obtenerse de los sucesos o eventos que ocurren en las transformaciones.

A fin de asegurar una adecuada percepción de los eventos considerados como más relevantes, acerca del acontecer de la transformación, y que a la postre servirán como base para dar cuenta del comportamiento real de ésta, es que se hace necesario llevar a cabo un proceso de modelado de la realidad de esta transformación en la perspectiva de los datos que la conforman. Sólo en esta perspectiva es posible lograr valores representativos y exactos para los vectores esenciales que requiere el regulador, puesto que un dato corresponde, en última instancia, al registro en algún medio de un evento o un suceso.

Es así entonces que surge la necesidad de un modelo de datos asociado a las transformaciones. Un *modelo de datos* es la herramienta que permite capturar tanto la dimensión estática, por medio de las estructuras, como la dimensión dinámica de la realidad, por medio de las operaciones que se aplican sobre tales estructuras.

Debe modelarse, por tanto, la transformación insumo-producto, a fin de establecer las estructuras de datos que mejor reflejen la dimensión estática de tal transformación. Del mismo modo, la dimensión dinámica, que se manifiesta en la ocurrencia de eventos, se retiene por medio de un conjunto de operaciones, de tal forma que cada evento específico que se suscite en la transformación tendrá una o más operaciones asociadas.

El proceso de modelado tiene como resultado un esquema, el cual define, en términos de estructuras y constricciones, es una *base de datos*.

En consecuencia, los eventos, que desean registrarse como datos, modificarán continuamente los estados de esta base de datos, a través del conjunto de operaciones establecidas en el modelo de datos.

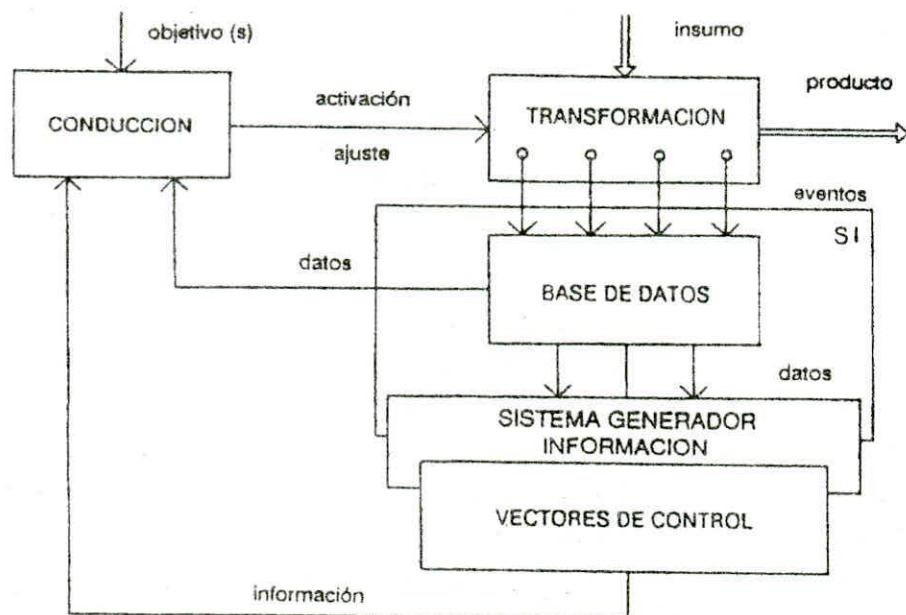
La base de datos se convierte, entonces, en una instancia instrumental necesaria entre los nodos de transformación y conducción.

Una vez plasmados los eventos como datos en la base de datos, se requiere luego de una instancia estructural capaz de transformar tales datos en información relevante y oportuna para apoyar las tareas decisorias que la conducción, en cuanto regulador, debe llevar a cabo, a fin de materializar las acciones de ajuste que se requieren para asegurar el logro de los objetivos. Aquella instancia corresponde a lo que se puede denominar *sistema generador de información*, y será un nodo inserto estructuralmente entre el nodo base de datos y el nodo de conducción. Aquel, en conjunto con las bases de datos, configuran los denominados *sistemas de información*.

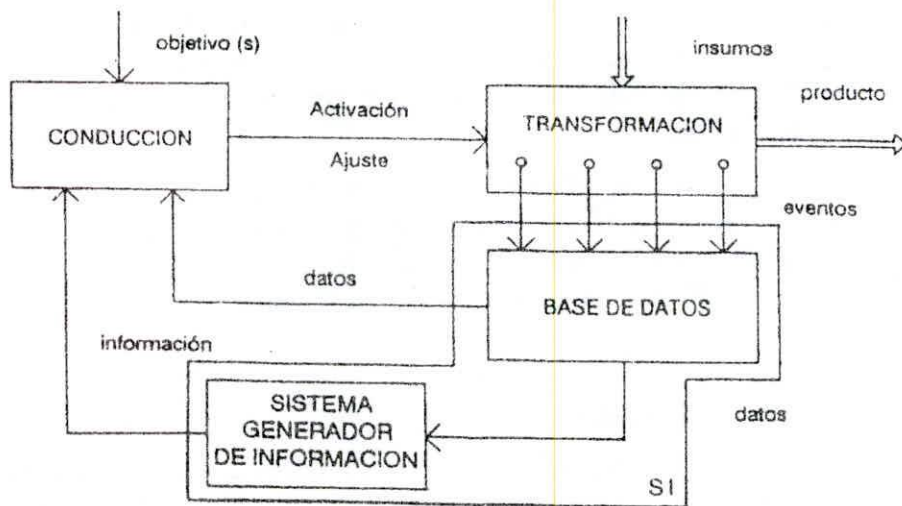
Por lo tanto, cualquier intento de comprensión de los procesos operacionales requiere del reconocimiento relacional de los tres nodos que lo conforman, es decir:

- conducción,
- transformación,
- sistema de información.

Ellos se relacionan tal como lo muestra la figura:



A objeto de facilitar el trabajo analítico emprendido, la figura anterior se simplifica de tal forma que en ella se destaca el rol de los tres componentes de los procesos operacionales desde una perspectiva estructural. El resultado es la siguiente figura:



1.16

6. Los Sistemas Informáticos en la Empresa.

Se puede decir que la *información* es aquella instancia que, al ser agente de disminución de incertidumbre, afecta el comportamiento de quien la recibe. En este sentido, y considerando que la incertidumbre representa un estado entrópico, los flujos de información serán, en consecuencia, flujos necesariamente negaentrópicos, puesto que reducen la incertidumbre de quien la recibe.

El valor de la entropía mínima corresponde siempre a quel nivel en donde la capacidad de absorber la variedad de la conducción llega al máximo. En consecuencia, la información proporcionada a la conducción debe ser tal que, junto con disminuir su incertidumbre, de acuerdo a la Ley de la Variedad Requerida de Ashby, mantenga la explicación del fenómeno y se encuadre dentro de aquella capacidad de absorber variedad.

En este contexto, la información, al interior de los procesos y en el marco de la Empresa como un todo, adquiere un sentido explícito toda vez que se constituye en un flujo que sustenta y facilita el logro de los niveles adecuados de efectividad, a los cuales todo nodo de conducción aspira, como agente de dirección de procesos, en cuanto a su capacidad de alcanzar con eficiencia y eficacia los objetivos específicos que la Empresa le ha encomendado.

En otras palabras, la información es uno de los componentes fisiológicos fundamentales de una Empresa, y su rol orgánico consiste en satisfacer los requerimientos específicos de las diferentes instancias de regulación o control. La información tiene su origen en los eventos que ocurren en los procesos orgánicos. Éstos deben ser observados, medidos, y registrados

como datos para luego transformarlos en la información que requieren los reguladores o controladores.

Si se analiza una Empresa desde el punto de vista del rol fisiológico de la información, es posible identificar una instancia orgánica en la cual ella es generada como tal para satisfacer los requerimientos de regulación o control. Ella corresponde a los subprocesos denominados *Sistemas de Información*.

En estos sistemas le cabe a la Informática un rol significativo en cuanto soporte tecnológico de procesamiento, razón por la cual se debe hablar, al hacer referencia a la categoría global en la cual se inscriben estos tipos de sistemas, de *sistemas informáticos*, entendiendo por tales a aquellos en los cuales existe participación relevante de la tecnología de computadores.

7. Qué es un Sistema Informático.

Uno de los propósitos que se persigue con la construcción de un sistema informático es su utilización como instrumento de apoyo al quehacer de la empresa, específicamente a una función o proceso en la cual han de insertarse con la pretensión de aumentar su efectividad, tanto en lo referente a las tareas operacionales como a las instancias decisionales.

En este contexto, hablar de un sistema informático es referirse a un producto que tiene diferentes características según la instancia empresarial que ha de apoyar. Si se trata de un sistema cuyo propósito es sustentar las tareas operacionales, ligadas a las transformaciones del tipo insumo-producto, se habla de *sistema de procesamiento de datos*.

Si el sistema informático está orientado a satisfacer las necesidades de información que se requiere en los procesos decisionales, se habla de *sistema de información*. Si estos últimos apoyan a la toma de decisiones estratégicas, se les conoce como *sistemas de información estratégicos*; por otra parte, si están ligados a las tareas de conducción de los procesos operacionales, se les conoce como *sistemas de información de gestión*, y en este caso, dado el carácter recursivo y jerárquico de los procesos operacionales, lo cual hace que los sistemas de información se agregen sinérgicamente, y bajo el criterio de información por excepción, el agregado superior de éstos se conoce como *sistema de información de gestión global*.

8. Los Sistemas de Procesamiento de Datos.

Los sistemas de procesamiento de datos no son en esencia sistemas de información, pues no necesariamente tienen un impacto en la tarea decisora del conductor; responden más bien a las necesidades de eficiencia y eficacia de las transformaciones, y generalmente son instancias de tratamiento de datos que apoyan a otras tareas, pudiendo incluso existir algunas transformaciones en las cuales todas sus tareas se llevan a cabo automatizadamente, como por ejemplo en los procesos de separación de compuestos en la industria del petróleo. En consecuencia, su propósito es aumentar la efectividad de las tareas operacionales, sobre todo aquellas que se caracterizan por lo complejo, lo repetitivo, y engorroso que implica el tratamiento de ciertos datos requeridos en la transformación.

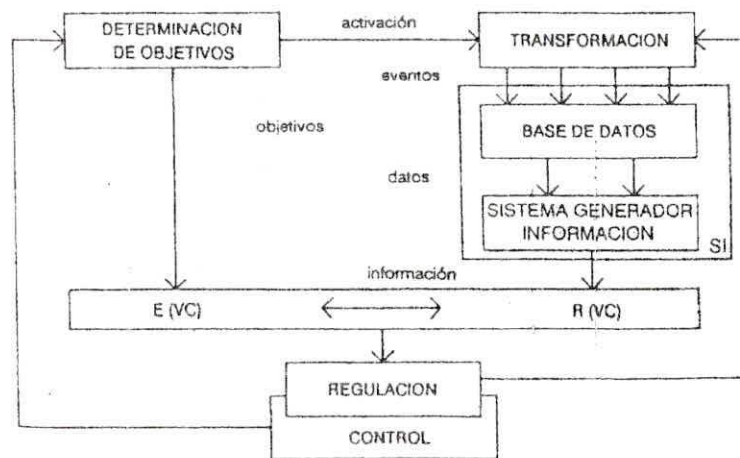
9. Los Sistemas de Información de Gestión.

Las instancias de conducción al interior del proceso deben tomar conocimiento de lo que acontece en la transformación que ellas dirigen. Ello implica observar las transformaciones a partir de los eventos que afectan a las variables de estados que las explicitan. Son estos eventos los que deben recogerse y registrarse en bases de datos, y que, mediante la transformación realizada por los sistemas generadores de información, se constituyen en los flujos de información que los nodos de conducción requieren.

Sin embargo, dadas las consideraciones anteriores, no todos los eventos deben ser recogidos ni deben sufrir tratamiento como datos, ya que podrían inundar con información superflua a la instancia de conducción, haciéndola perder eficacia. Por ello se sugiere utilizar los vectores de control como instrumento de determinación del flujo de información. Para hacerlos efectivos como instrumentos de control al interior de los sistemas de información, el conductor debe:

- Definir las variables componentes de cada vector de control que, como criterios de medición, deben dar cuenta de lo que acontece al interior de las transformaciones, en aquellos aspectos considerados como esenciales para el control, tanto en lo que se refiere al logro de los objetivos, como en cuanto a la capacidad operacional de los componentes del nodo de transformación.
- Determinar los valores esperados, que generalmente se expresan como rangos, para cada vector de control.

Los sistemas generadores de información, por su parte, deben recoger los datos mantenidos en las bases de datos, para determinar el valor real de cada vector de control operacional. De esta forma es posible comparar estos últimos con los vectores de valores esperados previamente establecidos, y obtener de este modo el vector de las desviaciones, para que, a partir de ellos, el nodo de conducción pueda regular la conducta de las transformaciones, a fin de ajustarla a la norma que el cumplimiento de los objetivos de actividad requiere. Este proceso se manifiesta en la siguiente figura:



Desde un punto de vista estructural, los sistemas de información (SI), a cualquier nivel, se constituyen en la instancia que al interior de cada proceso, relacionan las acciones operacionales realizadas con los nodos de conducción respectivos. Es decir, están

directamente asociados a los procesos de toma de decisiones, a partir de los cuales se materializa tanto la activación como el ajuste que el cumplimiento de los objetivos requiere, de acuerdo a las especificaciones finales de la Empresa.

A nivel de gestión, el rol de los sistemas de información consiste, como ya se señaló, en apoyar la tarea decisoria de la conducción del proceso a través de los mecanismos de activación y ajuste. En cuanto sistemas generadores de información (SGI), se les puede entender como una función que transforma datos en información, es decir,

$$\text{Información} = \text{SGI}(\text{datos})$$

Los datos corresponden al registro, en algún medio, generalmente una base de datos, de los eventos, razón por la cual son potencialmente informativos, esto es, que pueden convertirse en información si llegan a afectar el comportamiento de quien los recibe.

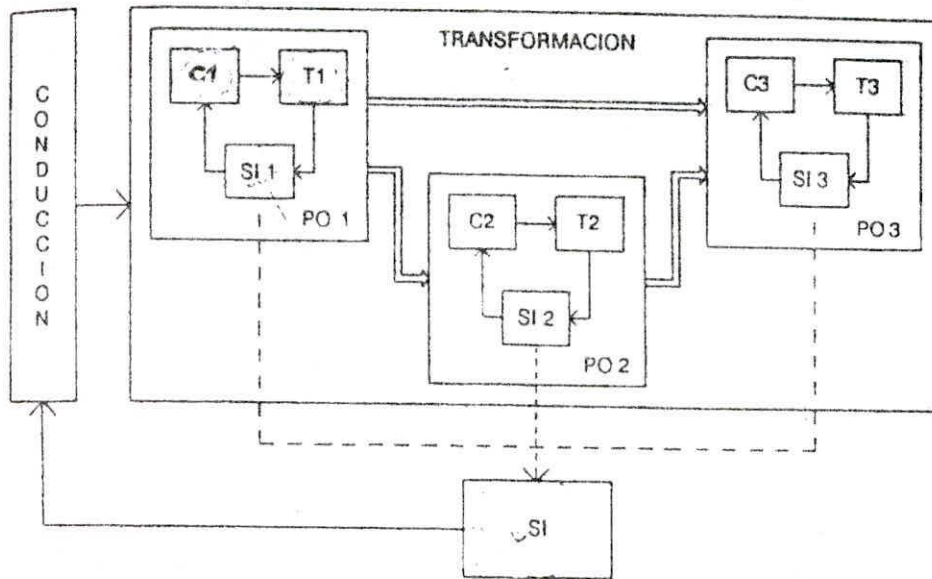
Se puede afirmar, entonces, que en el ámbito de los procesos que conforman el nivel de gestión, el input de los sistemas de información se origina en las transformaciones, donde los datos no son más que una expresión de los eventos que ocurren en ellas. Sin embargo, son los requerimientos de las instancias de conducción los que le confieren su status estructural.

Si se observa los niveles inferiores de la empresa, cada uno de los procesos manifiesta la existencia de sistemas de información que, apoyando el quehacer de la conducción, tienen como propósito dar cuenta de lo que acontece en las transformaciones, tanto en lo referente al logro de los resultados esperados, como en lo referente al estado de los componentes estructurales del proceso.

Esto significa que los sistemas de información se constituyen en un instrumento de control o regulación, ya que es a partir de ellos que la conducción puede alcanzar, a través del ajuste, la eficacia y eficiencia de la acción.

En esta perspectiva, son los vectores de control los que determinan el carácter, contenido y alcances de los sistemas de información, los cuales, por su parte, tienen su origen en los diferentes datos registrados en las bases de datos, los que, a su vez, se originan en los eventos que influyen en las variables de estado que determinan la conducta de la transformación.

Dado el carácter jerárquico y recursivo de la estructura organizacional, los sistemas de información en el nivel de conducción inmediatamente superior, se van agregando en los términos sugeridos por la siguiente figura, donde se ha omitido, por simplicidad, las bases de datos.



En este contexto, un sistema de información (SI) satisface los requerimientos específicos de la conducción a su nivel local, es decir, los de su proceso operacional (PO), pero a su vez también satisface, en lo que le corresponde, los requerimientos del nivel de agregación inmediatamente superior, respetando, por cierto, el criterio de variedad requerida.

En la figura anterior, el sistema de información SI1 satisface los requerimientos de la conducción del proceso operacional PO1; del mismo modo, el sistema SI2 aquellos del proceso PO2, y el SI3 los del PO3. Pero, a su vez, SI1, SI2, y SI3 conforman la base a partir de la cual se conforma el sistema de información SI.

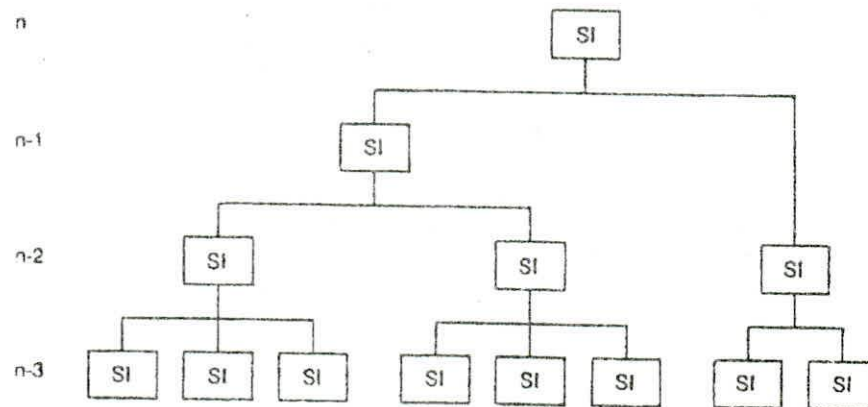
En este sentido, el problema de la información es un asunto de alcances globales, que involucra el funcionamiento fisiológico de la Empresa. Por ello, las soluciones informáticas locales, al no reconocer la verdadera dimensión del rol de la información en un contexto orgánico, no aseguran mejorar el rendimiento total, más aún, en algunas oportunidades la entorpecen al desequilibrar el funcionamiento de los componentes. En otras palabras, si bien las soluciones parciales resuelven algunos problemas, no son garantía de que sea lo más conveniente para la Empresa como un todo.

Ello se debe también, en alguna medida, a que las Empresas muchas veces son concebidas y desarrolladas sin explicitar debidamente la conformación estructural desde una perspectiva informacional. Generalmente aquellas se definen mediante un juego de jerarquías de autoridad, poder, y funciones, pero sin explicitar los flujos fisiológicos de información que toda Empresa requiere para su funcionamiento global.

Es por ello que se ha sugerido revisar las conformaciones estructurales, de tal forma de identificar los procesos componentes, recursivamente interrelacionados, en donde cada uno

de ellos explicita una tarea de conducción que garantice el logro de los objetivos, o subobjetivos asignados, marco en el cual la información juega un rol fundamental.

De esta forma, y centrando el análisis en los sistemas de información de cada proceso, se puede conformar una red de sistemas de información que, recursivamente, se van integrando desde abajo hacia arriba, en forma bottom-up, en los términos de la siguiente figura:



Cada sistema de información del nivel n-3 corresponde a una transformación en el nivel más bajo, es decir, a la última transformación que requiere de una instancia de conducción explícita. De esta forma, al estar integradas dichas transformaciones en un proceso de nivel superior, que a su vez es "conducido" como un todo, el sistema de información que lo sustenta corresponderá al agregado sinérgico de los sistemas de información del nivel inmediatamente inferior, y que se caracteriza por:

- La mantención de la variedad requerida.
- El criterio de excepción asociado a la autonomía relativa.

Ello significa que el nivel superior sólo recibe la información que da cuenta de aquellos eventos que escapan a la capacidad de resolución que tienen los niveles inferiores.

Si se continúa de esta forma hasta los niveles superiores, el nivel n corresponderá a aquellos sistemas de información integrados y que involucran el funcionamiento y la conducción del todo orgánico identificado como Nivel de Gestión.

Dicho nivel n, que se puede reconocer como Sistema de Información de Gestión Global, corresponde a aquel que además provee de los flujos de información interna que requiere el Nivel Estratégico.

10. Políticas Informáticas.

Se entiende por *política informática* al conjunto de guías o pautas de acción que conforman el marco de referencia, en virtud del cual se elaborarán los planes informáticos, y se enfrentará cualquier desarrollo informático posterior, tanto en lo referente a software como a hardware, como también en lo que se refiere a la función informática de la Empresa.

Las políticas informáticas difieren de una Empresa a otra, ya que normalmente se formulan para una entidad particular, pero en términos generales, en cualquier empresa no deben dejar de referirse a los siguientes tópicos:

- Definiciones en torno al hardware y software.
- Definiciones en relación a las prioridades.
- Definiciones en torno a la asignación de recursos.
- Definiciones en torno a los recursos humanos.
- Rol estructural y dependencia jerárquica de las unidades de informática.
- Centralización o descentralización de la función informática.

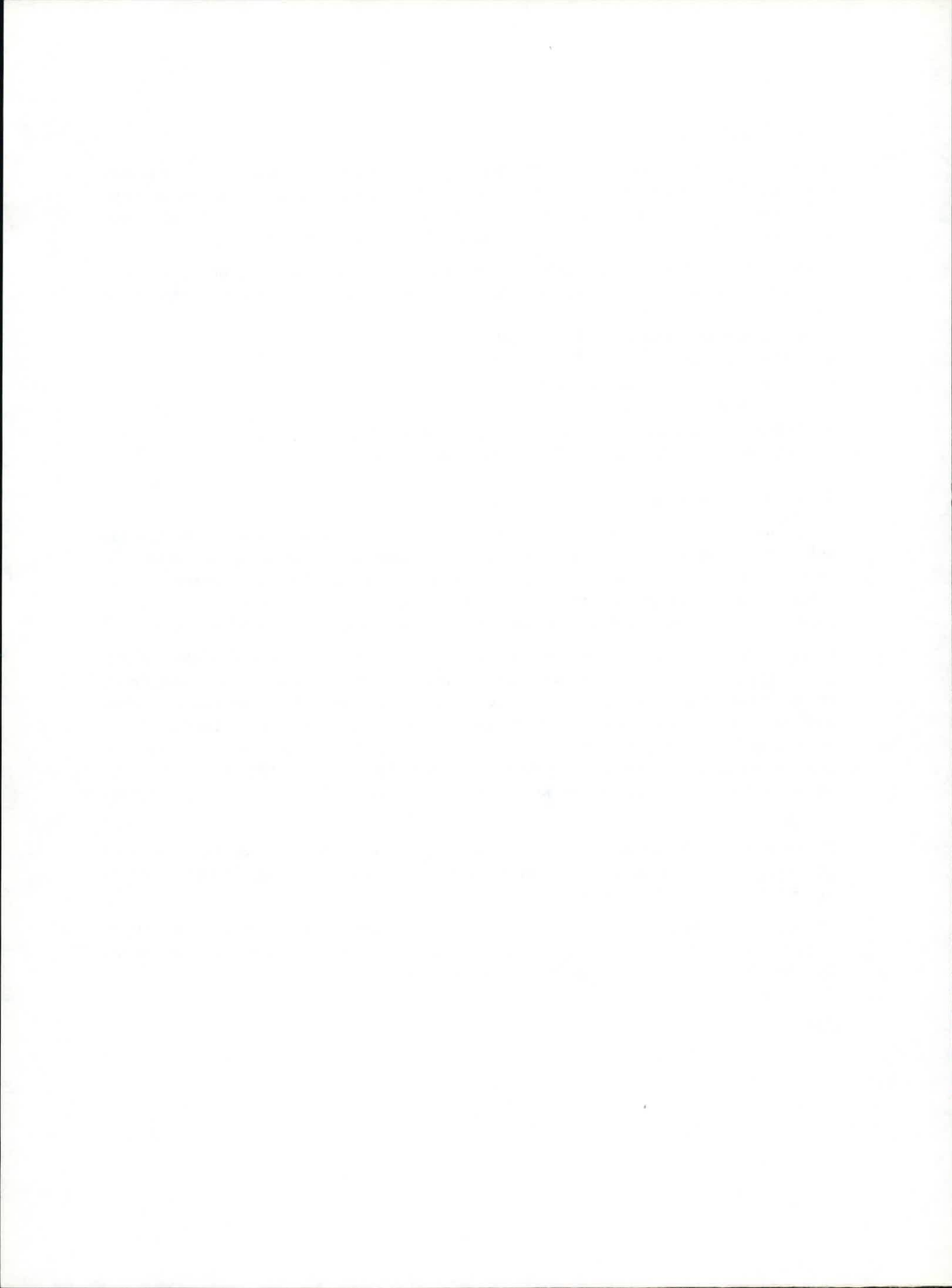
11. El Plan Informático.

Los *planes informáticos* corresponden a una distribución en el tiempo de las diversas aplicaciones (sistema de procesamiento de datos, sistemas de información de gestión, o sistemas de información estratégicos) susceptibles de desarrollar en el marco de una definición previa de las políticas informáticas. Se asocian también a ellos los recursos informáticos requeridos para llevar a cabo aquellas aplicaciones con posteridad.

Disponer de un plan a corto, mediano, o largo plazo, que guíe las actividades informáticas, presenta varias ventajas. Sobre todo cuando los sistemas llegan a ser más complejos, requieren más tiempo y recursos para ser desarrollados, utilizan archivos con datos comunes a diferentes sistemas, comprometen múltiples funciones organizacionales y tienen un mayor impacto en la gestión de la Empresa. Más aún, tener un plan informático implica contar con una base de comunicación entre los diferentes niveles de decisión de la Empresa, de modo de no entorpecer el desarrollo de sistemas por presión de las distintas prioridades individuales.

Por otro lado, tener un plan implica formalizar el valor que la Empresa asocia a los sistemas por desarrollar. Proporciona también un esquema para evaluar el desempeño logrado en las actividades de las unidades informáticas.

De acuerdo a estas consideraciones, el no tener un plan informático debidamente formulado, implicaría que el proceso de desarrollo de sistemas terminaría por perder toda relación con los planes y actividades de gestión de la propia Empresa.



II. DESARROLLO DE SISTEMAS INFORMÁTICOS.

El desarrollo de un proyecto informático es un proceso que tiene su origen en el reconocimiento de la existencia, en la Empresa, de una función que requiere la incorporación de ciertos niveles de automatización, y finaliza cuando, una vez redefinida, se inserta nuevamente en la estructura de ella.

Entre ambos momentos del desarrollo, es preciso cumplir un conjunto de tareas que incluyen: la definición del proyecto en cuanto a contenidos y alcances, el estudio de la posibilidad real de concluirlo con éxito, la identificación mediante un proceso de análisis de las subfunciones involucradas y la determinación de aquellas que habrá de automatizarse, el diseño de la estructura de los componentes automatizados, es decir, el software a construir, la construcción o programación del software, la habilitación final a objeto de ponerlo en operación con los niveles de eficacia preestablecidos, y por último, la incorporación de los cambios que la dinámica de la Empresa va exigiendo al sistema.

El desarrollo de un sistema informático debe tener siempre su punto de partida en lo establecido en los planes informáticos de la Empresa. Allí se estipula el momento en que ha de desarrollarse y las relaciones que manifiesta dicho sistema con el resto de los sistemas informáticos de la Empresa. En consecuencia, en virtud de lo indicado en el plan, en algún momento será preciso iniciar el desarrollo del proyecto, que ha de concluir con el sistema habilitado para cumplir su rol orgánico.

1. La Necesidad de una Metodología de Desarrollo.

En toda Empresa los sistemas de información, sean éstos estratégicos o de gestión, cumplen sin lugar a dudas con un rol protagónico, específicamente en lo que se refiere a la recolección, tratamiento, y entrega de la información relevante para la toma de decisiones.

Del mismo modo, los sistemas de procesamiento de datos, en virtud de los requerimientos de eficiencia y eficacia, se hacen cada día más imprescindibles, y su rol orgánico más determinante.

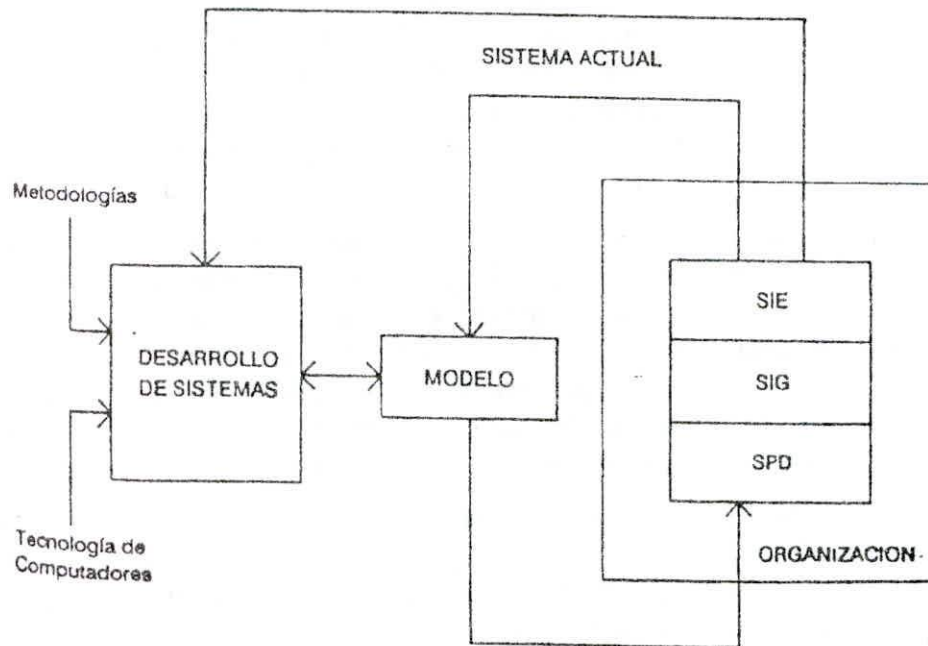
Construir y habilitar un sistema informático, ya sea que se trate de un sistema de información o un sistema de procesamiento de datos, es una tarea compleja por la multiplicidad de aspectos involucrados en ello, más aún cuando el proceso de desarrollo de un proyecto informático no se agota al momento de su habilitación, puesto que por el hecho de estar inserto en un entorno de alta variedad, está constantemente sometido a la necesidad de cambios para adaptarse a nuevas circunstancias.

En toda función o proceso, cualquiera sean sus características, siempre existen sistemas de información y de procesamiento de datos; sin embargo, la dinámica de la Empresa, el desarrollo tecnológico, y los avances en la comprensión del funcionamiento orgánico, presionan por su redefinición, por lo cual se hace necesario someterlos con cierta periodicidad a un proceso de desarrollo informático para incorporar los nuevos requerimientos, conformando de este modo un nuevo sistema.

Dado que no es posible trabajar con los procesos o funciones en su desenvolvimiento real actual, se hace necesario conformar todo nuevo sistema a partir de un trabajo basado fundamentalmente en el uso de modelos; por lo tanto, el desarrollo informático del sistema exige del modelamiento del sistema actual para, una vez incorporados los nuevos

requerimientos, conformar un modelo del sistema redefinido, el cual ha de introducirse nuevamente a la función o proceso correspondiente.

En consecuencia, el desarrollo informático del sistema se realiza fundamentalmente en torno a un modelo que debe basarse en un cierto planteamiento metodológico, y dado que lo que se pretende en última instancia es la incorporación de ciertos niveles de automatización, se hace necesario, como un aspecto significativo, considerar la capacidad potencial de la tecnología de computadores. Esto se puede representar en la siguiente figura:



2. Las Bases para una Metodología Efectiva.

Todo proyecto informático, ya sea que se trate de sistemas de información estratégicos, de sistemas de información, o de sistemas de procesamiento de datos, debe ser sometido a un proceso de desarrollo que tiene su origen en la necesidad que tiene toda función de mejorar su aporte al funcionamiento global de toda la Empresa. El aumento de los niveles de automatización es un factor de principal importancia al respecto. Por lo tanto, ese proceso de desarrollo normalmente ha de conducir a la obtención de un producto de software que ha de insertarse en la trama estructural del proceso en el cual deberá cumplir su rol funcional.

Sin embargo, el logro de tal producto, a partir de aquella necesidad, no es asunto trivial. Más aún, la cantidad de factores involucrados, así como su diversidad, exigen de un ordenamiento tal que permita la obtención de resultados, tras un trabajo eficiente y eficaz.

En consecuencia, el contar con una metodología adecuada es un requisito fundamental para la tarea de desarrollo de un sistema que contiene como componente fundamental un producto de software.

La metodología debe descansar en ciertas premisas básicas, tales como la posibilidad de obtención de resultados parciales, a objeto de corregir y realimentar antes de obtener el producto final, lo cual implica modularidad.

No pocas metodologías se caracterizan por el largo período de tiempo que toma el desarrollo del sistema. Este tiempo transcurrido es función del tamaño del sistema y no es poco usual que tome incluso años. Con toda seguridad puede ser reducido, realizando algunas subetapas en paralelo, o mejorando los instrumentos requeridos para conformar un sistema eficiente y eficaz, pero para que ello sea posible debe ser concebido y planeado muy cuidadosamente, pues de lo contrario puede dar origen a inconsistencias y requerir, contra lo deseado, más tiempo, en vez de menos.

Uno de los principales problemas en el desarrollo de un sistema se origina en las primeras fases, al tratar de determinar qué es lo que desea el usuario, pues éste usualmente no está acostumbrado a escribir en suficiente detalle lo que desea, como para modelar con exactitud un proceso organizacional actualmente en función, o bien un sistema a implementar. No hay duda que, idealmente, es muy importante iniciar el desarrollo de los sistemas de acuerdo a los requerimientos correctos. Diseñar, implementar, y poner en operación un sistema para satisfacer requerimientos no deseados es obviamente una pérdida de tiempo y esfuerzo.

Un sistema debe ser concebido, diseñado, y construido para aumentar la eficiencia organizacional. Es necesario entonces que los altos ejecutivos estén involucrados en el desarrollo de los sistemas informáticos; ello es posible mediante la incorporación de las políticas en informática a las políticas generales de la Empresa, de modo que a partir de ellas se puedan establecer las responsabilidades de los especialistas y los interesados directamente en el desarrollo de un sistema, y a la vez asignar los recursos y las prioridades cuando aquellos sean escasos.

3. Un Planteamiento Metodológico para el Desarrollo de un Sistema Informático.

El enfoque que utiliza la Empresa para enfrentar el proceso de desarrollo de un sistema, tendiente a la redefinición de una función o proceso organizacional a la cual está asociado, dependerá fundamentalmente de sus políticas informáticas.

Enfrentar un problema informático y encontrar su solución requiere siempre de un enfoque global, de un método en lo específico, y de un conjunto de herramientas para alcanzar soluciones eficientes y eficaces.

Dado que todo sistema informático es una componente orgánica, por cuanto forma parte de un subsistema organizacional, entonces cualquier metodología de desarrollo debe supeditarse necesariamente a esta condicionante. Es decir, cualquier planteamiento

metodológico debe considerar como punto de partida o como referente el hecho que el sistema corresponde a una función o proceso de la Empresa.

En general, y a modo de una primera aproximación, se puede decir que metodológicamente el desarrollo de un sistema de este tipo comprende tres grandes fases:

- Una fase de definición y análisis del problema.
- Una fase de diseño e implementación.
- Una fase de funcionamiento rutinario.

Estas fases corresponden a una visión muy general de la forma de resolver el problema, la cual no permite comprender los verdaderos alcances de las diferentes tareas involucradas y, desde un punto de vista metodológico, no se constituyen en una guía rigurosa y efectiva, que sea capaz de conducir realmente aquel proceso de desarrollo.

Ello pone de relieve la necesidad de desagregar aquellas tres fases, para conformar una estructura que permita llevar a cabo un desarrollo sistemático, capaz de conferir al trabajo niveles adecuados de eficiencia y eficacia.

Dicha estructura de fases nunca será absoluta, y la conformarán una mayor o menor cantidad de etapas, dependiendo de las características del medio en que habrá de aplicarse. Entre estas características cabe destacar:

- Tamaño de la organización.
- Tamaño, complejidad, y duración del proyecto.
- Necesidad de coordinar el trabajo de los distintos especialistas.
- Necesidad de establecer puntos de control adecuados.
- Necesidad de particionar la documentación.
- Necesidad de agrupar lógicamente los recursos.
- Necesidad de ubicar grupos de actividades con intereses más bien comunes.

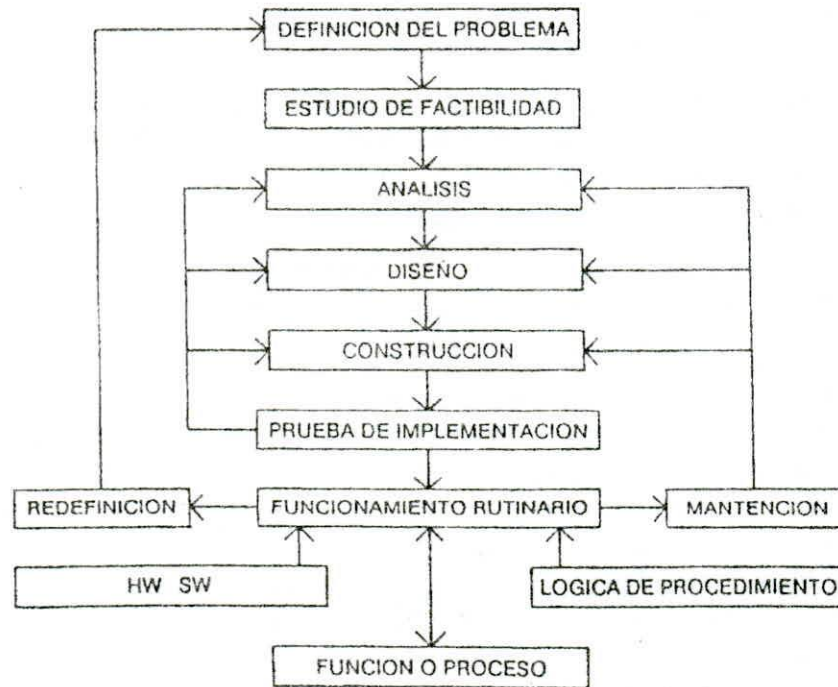
No obstante ello, es posible conformar un cuerpo metodológico susceptible de ser utilizado con eficacia y eficiencia en cualquier proceso de desarrollo de un sistema informático, cualquiera sea la Empresa que lo requiera.

La estructura metodológica sugerida se basa fundamentalmente en el establecimiento de un conjunto de etapas por las cuales ha de transitar cualquier proyecto informático. Ella está basada en los planteamientos de D. Teichroew, publicados en su trabajo "Improvements in the System Life Cycle".

A partir de lo sugerido por Teichroew, se puede conformar una estructura metodológica apropiada para el desarrollo de un proyecto informático, que consta de ocho etapas:

- Definición del problema.
- Estudio de factibilidad.
- Análisis.
- Diseño.
- Construcción.
- Prueba e implementación.
- Funcionamiento rutinario.
- Mantención.

La secuencia e iteración de estas ocho etapas aparece en la siguiente figura:



3.3

4. Definición del Problema.

Esta fase se refiere esencialmente a la especificación del sistema que se pretende desarrollar.

Existen diversas fuentes a través de las cuales es posible detectar la necesidad de proveer a una función organizacional de una solución informática. Entre ellas cabe señalar: la inquietud de algunos ejecutivos de la empresa, las necesidades de aquellos que están involucrados operacionalmente en la función, así como también las sugerencias de los especialistas en Informática. Sin embargo, lo más acertado es que ella surja de lo señalado en el plan informático vigente en la organización, el cual referencia la distribución temporal de todos los sistemas informáticos que se requiere desarrollar.

Por lo tanto, la definición del problema tiene su punto de partida en el reconocimiento explícito de la existencia de un sistema, es decir, de una función o proceso empresarial que requiera de una solución informática, y su propósito es especificar, por una parte, la naturaleza, contenido, objetivos, y restricciones del sistema que se pretende desarrollar, así

como el entorno en el cual se encuentra inserto, y por otra, los cambios que se desea incorporar en la nueva conformación de la función.

La definición del problema consiste entonces en obtener una primera aproximación al sistema a desarrollar, entendiéndolo como un problema de la Empresa, ligado a una función o proceso, que requiere de una solución informática.

En este sentido debe hacer referencia a dos aspectos fundamentales:

- La función y su entorno.
- La dimensión del cambio deseado.

Para ello es preciso:

- Definir la función objeto de estudio identificando y caracterizando su entorno y las subfunciones que lo conforman.
- Establecer los alcances de las soluciones deseadas por los involucrados en el quehacer inherente a la función.

5. Estudio de Factibilidad.

Todos los proyectos son factibles de realizar teniendo recursos ilimitados y tiempo infinito. Desafortunadamente el desarrollo de un sistema apoyado por computador está sujeto a ciertas restricciones. Entre ellas se encuentran los recursos financieros, que son una limitante significativa, que trae como consecuencia, muchas veces, la imposibilidad de contar con la tecnología que la solución al problema requiere. La capacidad operacional que existe en la Empresa y los recursos tecnológicos requeridos para la generación de información y el procesamiento de datos, son también un factor de limitancia que es preciso considerar.

Los proyectos de desarrollo de sistemas tendrán un verdadero éxito si permiten a un área o función de la Empresa obtener mejoras que se traduzcan directa o indirectamente en beneficios significativos. De esta manera, si un sistema provee de información para aumentar la eficacia de la toma de decisiones, facilita la incorporación de características deseables a un producto, o incluso disminuye costos y gastos administrativos, entonces ese sistema sí se justifica, pero previamente se debe llevar a cabo un análisis de factibilidad que demuestre tales bondades.

6. Análisis.

En el ámbito del desarrollo de un proyecto informático, el análisis consiste en identificar y especificar el conjunto de subfunciones orgánicas que han de conformar el sistema de información. Para ello, es preciso identificar la estructura de la función sometida a tratamiento informático, es decir, descomponerla en subfunciones e interfaces, y luego estudiar su composición y determinar los cambios estructurales necesarios, incorporando, eliminando, agrupando, o separando las subfunciones, para así establecer, a continuación, cuáles de ellas serán automatizadas y cuáles serán manuales, con el propósito de reinsertarla nuevamente en la Empresa una vez concluida su construcción e implementación, ahora redefinida estructuralmente y con nuevos niveles de soporte automatizado para el tratamiento de datos y la generación de información.

Por lo tanto, el propósito de esta etapa es generar una especificación funcional que describa la función o proceso objeto de desarrollo informático. En ella se debe hacer referencia a los conjuntos de datos que debe procesar la función desde que aquéllos ingresan a ella, desde una función relacionada de entrada hasta una función relacionada de salida, pasando por una red de subprocesos intermedios, y que corresponden a la conformación estructural de la función.

Se hace referencia en aquella especificación también a las interfaces entre los subprocesos, es decir, a las estructuras de datos subyacentes en cada uno de los conjuntos de datos que debe procesar cada subfunción, y a los procedimientos que han de regir en ellas las transformaciones de los flujos de datos de entrada en flujos de datos de salida que debe realizar.

Especifica, además, los requerimientos de tratamiento automatizado, es decir, el conjunto de subprocesos definidos como aquellos que realizarán sus tareas mediante procesamiento automatizado y los procedimientos requeridos para el manejo de las interfaces entre las subfunciones automatizadas y las subfunciones regidas por procesos manuales.

En consecuencia, las actividades a cumplir durante la fase de análisis tienen que ver directamente con la función objeto de estudio y en una primera aproximación consisten en:

- Definir el ámbito del sistema, es decir, sus relaciones con las otras funciones a las cuales se encuentra ligada inmediatamente.
- Describir la conformación actual de la función, es decir, los subprocesos que la conforman y los flujos de datos que los relacionan.
- Introducir los cambios necesarios para una eventual nueva concepción de la función o proceso, estableciendo incluso una nueva configuración lógico-estructural si la efectividad de la Empresa así lo requiere.
- Determinar algunas alternativas en relación al cómo llevar a cabo las subfunciones, estableciendo como elemento diferenciador el grado de automatización asignado a cada uno de ellos.
- Realizar un estudio costo/beneficio para cada alternativa.
- Seleccionar la alternativa más ventajosa.
- Formalizar por escrito la especificación del nuevo sistema, es decir, establecer los procedimientos que han de regir las transformaciones que deben realizar las subfunciones establecidas como manuales y una especificación que explicita los requerimientos de procesamiento automatizado para el producto de software a construir.
- Estimar el desarrollo posterior del sistema para elaborar un programa, y el correspondiente presupuesto.

En la historia del desarrollo de los proyectos informáticos es posible distinguir varios modos de enfrentar el análisis del sistema: el Análisis Clásico, el Análisis Estructurado, y el Enfoque de Objetos.

El *Análisis Clásico* se caracteriza fundamentalmente por carecer de herramientas que permitan reflejar lo logrado por el proceso de análisis de la función o proceso. Sus especificaciones son, esencialmente, narrativas, lo cual las hace poco rigurosas, ambiguas, rígidas, y de difícil mantención, a la vez que no se constituye en un instrumento de comunicación eficaz entre el analista y el usuario.

El *Análisis Estructurado* es un enfoque para enfrentar la fase de análisis en el desarrollo de un sistema informático que se basa fundamentalmente en el uso de modelos para representar el sistema. Estos modelos a su vez descansan en la utilización de la noción de estructura jerárquica. El Análisis Estructurado comienza con la descripción de la estructura actual del sistema, luego incorpora algunas modificaciones a la función, tanto en lo referente a su conformación estructural, como en cuanto al grado de utilización del procesamiento automatizado de datos o información, a objeto de lograr mayores niveles de eficacia y eficiencia.

El *Enfoque de Objetos* comienza identificando las componentes estructurales del sistema; para cada una de ellas define sus variables de estado y las operaciones que pueden cambiar esas variables, y las encapsula en un bloque al que denomina *objeto*. Así, el sistema se ve en forma natural como una sociedad de objetos que interactúan a través de interfaces y mecanismos de comunicación. Los sistemas desarrollados bajo este enfoque son muy robustos y eficientes.

Los métodos más utilizados en la actualidad son los dos últimos, vale decir, el Análisis Estructurado y el Enfoque de Objetos.

7. Diseño.

El problema del diseño de un sistema informático comienza cuando ya han sido evaluadas, caracterizadas, y documentadas todas aquellas necesidades de tratamiento automatizado de datos para generar otros datos al interior del sistema que se desarrolla, tarea que compete a la fase de análisis. Sin embargo, debe tomarse en cuenta que las actividades de análisis y diseño son iterativas, lo cual implica que algunos aspectos del análisis pueden ser modificados, debido a consideraciones que surgen durante la fase de diseño.

El diseño tiene como propósito la generación de alternativas de solución al problema de cómo manejar y procesar automatizadamente los datos para generar otros datos o información, en el marco de una determinada configuración computacional.

Llevar a cabo el diseño de un sistema requiere del conocimiento de las técnicas para configurar productos de software, tanto a nivel de conformaciones estructurales, como a nivel de los procedimientos que han de guiar la tarea de cada uno de los componentes de aquellas conformaciones, así como también del conocimiento de las técnicas para organizar los datos que debe manejar el sistema. El hecho de requerir una determinada configuración computacional exige la necesidad de tener un conocimiento profundo respecto a las capacidades de los equipos, de sus facilidades, de sus modos de operación, y de las formas de organizar y estructurar los datos.

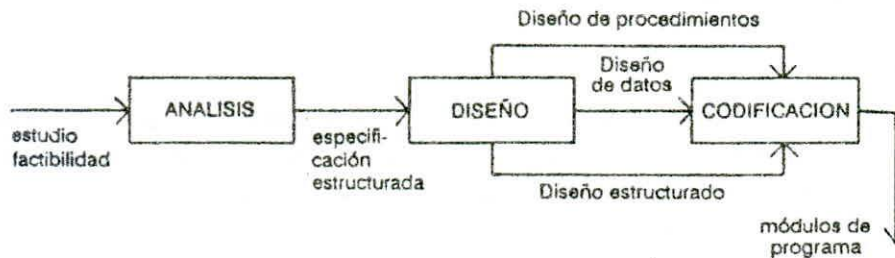
El diseño debe permitir configurar un producto de software que tenga como premisas básicas la facilidad de su construcción y su posterior mantención eficiente, una vez que entra en funcionamiento normal.

En lo específico, la fase de diseño tiene como propósito fundamental especificar la estructura y procedimientos que conforman el software, es decir, las características e interrelaciones de los elementos involucrados en el sistema que ha de someterse a procesamiento computacional. Estos elementos deben ajustarse a los requerimientos

definidos en la fase de análisis, y ajustarse a las normas y restricciones que impone la Empresa a través de las políticas informáticas globales y aquellas específicas referidas al desarrollo y construcción de un producto de software, y satisfacer también los criterios de evaluación predefinidos.

En una perspectiva general, el diseño, como actividad humana, puede caracterizarse como el proceso de aplicar diferentes técnicas y principios con el propósito de definir un dispositivo, proceso, o sistema, con el propósito de generar el modelo de un sistema que ha de someterse posteriormente a construcción.

En el ámbito de la Informática, el diseño es una etapa del desarrollo de un sistema que consiste de la generación de las estructuras de datos, y de la estructura y procedimientos del producto de software que ha de satisfacer los requerimientos de procesamiento automatizado de datos, contenidos en la especificación lograda en la fase de análisis, tal como se muestra en la siguiente figura:



La especificación estructurada o funcional muestra, desde el punto de vista de los datos, los diferentes subprocesos o subfunciones que conforman el nuevo sistema propuesto, explicitando el modo en que aquellas se llevarán a cabo, distinguiendo, en consecuencia, entre las que se llevarán a efecto de manera manual y las que se realizarán de manera automatizada.

8. Construcción del Sistema.

La etapa de construcción de un producto de software tiene relación con la producción de los diferentes programas de aplicación que se han especificado en la etapa de diseño. Estos resultados se deben alcanzar dentro de las restricciones que la Empresa le establece al proyecto, y por lo tanto es preciso tratar de buscar la maximización de la función objetivo especificada tanto para el desarrollo como para la operación normal de los programas.

En lo específico, la construcción de sistemas se apoya en: las especificaciones generadas en la etapa de diseño; los estándares establecidos respecto de los lenguajes a utilizar, modelos de documentación, las técnicas de programación, etc.; las características de operación de los

equipos utilizados para la ejecución de los programas, los sistemas operativos y software básico disponibles, entre otros aspectos.

La construcción se traduce en última instancia en el conjunto de programas funcionando en el ámbito del sistema, acompañados de la documentación correspondiente, es decir, los listados de los programas y sus resultados, y los respectivos manuales de operación.

El pasar de un diseño detallado a un lenguaje de programación corresponde a una parte importante dentro de lo que se denomina Ingeniería de Software. Cabe hacer notar que este paso se puede realizar de diversas maneras. Es muy probable que una interpretación inadecuada del diseño detallado produzca errores en la conformación del código fuente. La complejidad de los lenguajes de programación o sus restricciones pueden, a menudo, dar origen a códigos fuentes con tales grados de confusión que hacen ineficientes y difíciles de mantener a los programas; más aún, las características de un lenguaje de programación pueden influir imponiendo limitantes en forma innecesaria al diseño de software y a las estructuras de datos.

En la actualidad existe el convencimiento de que el software diseñado debe responder a ciertos cánones de calidad. Es así como ahora se habla de la *calidad del software*, concepto que abarca características tales como claridad, portabilidad, facilidad de comprensión, modificación y prueba, modularidad, y facilidad de mantención.

Es conveniente entonces emplear técnicas de programación que faciliten la construcción de un software de alta calidad. La programación estructurada y la programación de objetos son las técnicas que en la actualidad más se adecúan a esa perspectiva.

9. Prueba e Implementación.

La implementación es el proceso mediante el cual un sistema informático se transfiere desde los especialistas que lo han desarrollado a los responsables de su operación, a objeto de incorporarlo definitivamente a la dinámica estructural de la Empresa.

En consecuencia, la implementación consiste en la entrega del sistema al usuario para su aprobación final, con miras a su incorporación definitiva a las tareas operacionales o de gestión que la función o proceso a la cual corresponde debe llevar a cabo, en el marco de una organización en funcionamiento. Este proceso de incorporación debe abordarse tanto desde el punto de vista de los procedimientos relativos a los aspectos manuales, como de los procedimientos relativos al tratamiento automatizado de los datos, por cuanto son todos ellos en conjunto los que permitirán al sistema cumplir su rol organizacional.

Por esta razón, antes de incorporar el sistema a la estructura de la Empresa a nivel de implementación, es preciso someter el producto de software, entendido éste como un componente estructural de la función o proceso orgánico, a las pruebas necesarias para establecer su efectividad como instancia de procesamiento automatizado de datos, y luego llevar a cabo los procesos de conversión de datos que el nuevo software requiere.

10. Funcionamiento Rutinario.

El funcionamiento rutinario es aquella etapa del desarrollo de un sistema en la cual éste se inserta en la dinámica de la Empresa en forma definitiva, ahora redefinido y habiendo introducido en su conformación estructural algunas modificaciones en relación a cómo se llevarán a cabo, e incluso algunos cambios respecto del quehacer del sistema. En otras palabras, consiste en la puesta en funcionamiento de las diferentes subfunciones, algunas manuales y otras automatizadas, que conforman el sistema, de manera que se cumpla con los requerimientos para los cuales fue desarrollado dentro de los factores de desempeño que le fueron asignados.

En términos más concretos, el funcionamiento rutinario debe orientarse a lograr que el sistema de información opere de manera eficiente y eficaz, para lo cual es preciso evaluar periódicamente el desempeño tanto a nivel de procedimientos como de recursos disponibles, manteniendo actualizados los parámetros considerados en el caso costo/beneficio llevado a cabo para evaluar la efectividad del sistema.

Para lograr dicho funcionamiento en forma eficiente es preciso establecer los mecanismos de control necesarios para corregir los errores que se produzcan y, en general, para mejorar el funcionamiento del sistema.

El funcionamiento rutinario exige de ciertas tareas ineludibles, tales como la preparación de estándares, para llevar a cabo el control de la operación del sistema; la preparación de procedimientos administrativos que permitan la detección y análisis de errores, y el establecimiento e implementación de procedimientos estándares que faciliten el control de la evolución del desempeño del software en relación a su rol organizacional.

11. Mantenimiento del Sistema.

La mantención es una de las actividades que más recursos insume en la vida de un sistema. Aunque los costos globales de la mantención han bajado estos últimos años como producto de la incorporación de nuevas técnicas de análisis y diseño, lo cierto es que ocupa por lo menos el 50% de los recursos del proyecto, lo cual la convierte en una etapa crucial en la vida del sistema. Sin embargo, la mantención aún es un proceso difícil de encarar por parte de las unidades de Informática de la Empresa. La información estadística muestra que sólo el 23% de las tareas pendientes son nuevos desarrollos, el 27% cambios relevantes, y el 50% restante son cambios menores. Aún cuando los recursos que absorbe la mantención sean menores, mejorando por consiguiente la utilización de los recursos disponibles, un nuevo desarrollo y los problemas derivados de un número tan importante de tareas obligan a gastos de gran envergadura.

En lo que respecta al sistema como un todo, la mantención consiste en observar continuamente la operación del sistema y el desempeño de sus diferentes componentes, de modo de poder realizar las modificaciones que se necesiten para mantener el desempeño del sistema dentro de los límites establecidos al determinar los requerimientos. Esto puede requerir tanto la modificación del sistema, como incluso su rediseño total.

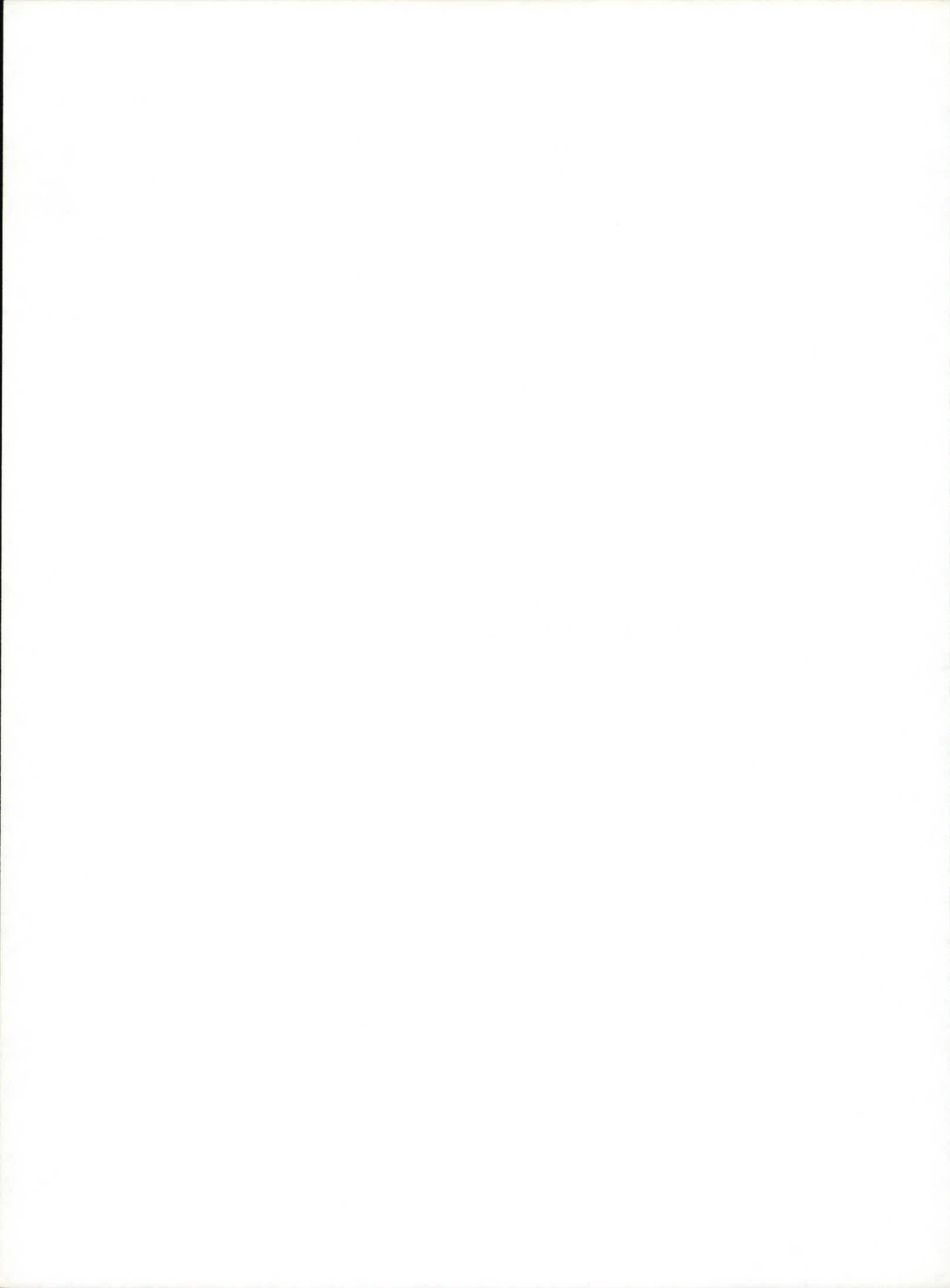
Para llevar a cabo la mantención es preciso contar con los requerimientos que exigen adecuar el sistema a ciertos cambios producidos, tanto en los objetivos como en los

procedimientos, en virtud de los cuales el sistema funciona. El origen de estos requerimientos se puede encontrar en ciertos cambios producidos en la configuración del software y el hardware que soportan el sistema, en la detección de errores en la lógica o en las actividades de manipulación de datos del sistema, en las variaciones en las características sobre las cuales se basó el diseño del sistema, y por último, en los nuevos requerimientos del usuario.

Normalmente la dinámica del funcionamiento de la Empresa impone modificaciones a las diferentes funciones o procesos, consistentes ya sea en la incorporación, eliminación, o modificación de algunas de ellas, lo cual implica de algún modo una redefinición del sistema.



III. ANÁLISIS Y DISEÑO ESTRUCTURADO.



El análisis de sistemas

La palabra "análisis" significa descomponer. Su origen etimológico está en la palabra griega "analysis" que significa, justamente, "descomposición de un conjunto en partes".

Este es el sentido que, en un marco de referencia amplio, asumen las tareas que, dentro del desarrollo de un proyecto informático, le corresponde realizar a la fase de análisis.

El análisis consiste en identificar y especificar el conjunto de subfunciones orgánicas que han de conformar el sistema de información o el sistema de procesamiento de datos. Para ello es preciso identificar la estructura de la función sometida a tratamiento informático, es decir, descomponerla en subfunciones e interfases y luego estudiar su composición y determinar los cambios estructurales necesarios incorporando, eliminando, agrupando o separando subfunciones para así establecer, a continuación, cuáles de ellas serán automatizadas y cuales serán manuales, todo ello con el propósito de reinsertarlas nuevamente en la Organización una vez concluida su construcción y su implementación, ahora redefinida estructuralmente y con nuevos niveles de soporte automatizado para el tratamiento de datos y la generación de información.

Por lo tanto, la fase de análisis, dentro del desarrollo de un proyecto informático, consiste en la obtención de una especificación exhaustiva de una cierta función, la cual, tras un proceso de redefinición, tanto en lo que se refiere a su conformación orgánico-funcional como en relación a la forma de llevar a cabo los procedimientos que rigen las tareas operacionales que le son propias, se debe incorporar a la dinámica organizacional en la cual se encuentra inserta.

Esta especificación, además de corresponder a una descripción detallada del sistema, explicita los requerimientos de procesamiento automatizado que la redefinición del sistema exige.

Para lograr aquello, la fase de análisis metodológicamente requiere, en primer término, describir la estructura y la forma en que actualmente se llevan a cabo las subfunciones o subprocesos de la función sometida a análisis.

A partir de esta descripción se redefine la conformación estructural de la función, incorporando, eliminando o desagregando subfunciones de tal modo que una vez estructurada en virtud de estos cambios, sea posible establecer el grado de automatización a incorporar.

Esta tarea consiste en decidir y establecer cuál será la mejor manera de emplear la tecnología de procesamiento de datos para llevar a cabo las tareas de la función, en cuanto ya sea al manejo de datos o generación de información, en términos tales de lograr los mejores niveles de eficiencia y eficacia. Para ello, es preciso en primer término, sugerir alternativas en función de los grados de automatización incorporados, para luego, de acuerdo a un criterio de costo-beneficio, seleccionar la mejor opción.

La fase de análisis concluye con la documentación de una especificación funcional en la cual se describe la función o proceso sometida a tratamiento informático. Esta descripción hace referencia a los conjuntos de datos y a los subprocesos que los afectan desde que aquellos arriban a la función, provenientes de una cierta función relacionada de entrada, hasta que la abandonar dirigiéndose a una función relacionada de salida.

Describe también las interfases entre los subprocesos, es decir, las estructuras de datos subyacentes en cada uno de los conjuntos de datos que debe procesar cada subfunción, y los procedimientos que han de regir en ellas las transformaciones de los flujos de datos de entrada en flujos de datos de salida.

Especifica, además, los requerimientos de tratamiento automatizado, es decir el conjunto de subprocesos definidos como aquellos que realizan sus tareas mediante procesamiento electrónico, y los procedimientos requeridos para el manejo de las interfaces entre las subfunciones automatizadas y las subfunciones regidas por procesos manuales.

4.1. Evolución de las metodologías de análisis

Uno de los planteamientos metodológicos para enfrentar el Análisis de Sistema que adquiere mayor relevancia en los últimos años es el denominado Análisis Estructurado.

Los orígenes del Análisis Estructurado con toda seguridad se encuentra en la programación Estructurada.

A mediados de los años 60 surgen los trabajos de Bohm-Jacopini [BOH-62] quienes con su famoso teorema demuestran que cada programa por complejo que sea puede desglosarse en tres estructuras: secuencia, selección, a la que corresponde la construcción IF-THEN-ELSE, e iteración, asociada a la construcción DO WHILE. En 1968, Edgar W. Dijkstra [DIJ-68] proclama en una carta enviada a la revista ACM la necesidad de eliminar la sentencia GOTO en todos los lenguajes de alto nivel.

Todas estas propuestas van conformando los cánones de la programación estructurada cuyos fundamentos son:

- Definir una serie de construcciones estándar sobre las que articular por bloques la totalidad del programa.
- Subdividir un problema complejo en subproblemas más sencillos, obteniendo conformaciones del tipo top-down para los programas.
- Determinar la topología y las estructuras de datos adecuadas para el problema considerado.

En esta misma línea, entre 1968 y 1972, Edward Yourdon y Larry Constantine [YOU-78] dan cuerpo y contenido al Diseño Estructurado que se basa fundamentalmente en dos características, la estructura jerárquica de los componentes procedurales y la estructura de los datos y la modularidad. La estructura del software se deriva mediante un proceso de partición, que relaciona los elementos de una solución de software con partes de un problema del mundo real definido explícitamente durante el análisis de requerimientos.

La modularidad es un enfoque que permite reducir la complejidad, facilita los cambios, es decir, la mantención del sistema y da como resultado una más fácil implementación posibilitando el desarrollo paralelo de diferentes partes del sistema.

Sin embargo, a pesar de los avances significativos en la construcción y posterior mantención de productos de software, persistía el problema de la conformación de sistema, el cual habitualmente no se ajustaban exactamente a los requeri-

mientos organizacionales, en otras palabras, se diseñan, construyen, implementan y ponen en ejecución sistemas bien diseñados y contruidos, pero lamentablemente equivocados. El origen de esta situación radica en las limitaciones de la especificación de los requerimientos del sistema, producto principal de la etapa de análisis.

Por estas razones a mediados de la década del 70 Douglas Ross, Edward Yourdon y Tom de Marco [ROS-77][ROS-85] desarrollan las ideas fundamentales de lo que se ha venido a denominar Análisis Estructurado, como respuesta a las limitaciones de aquella forma metodológica de enfrentar estas tareas denominadas corrientemente Análisis Clásico.

4.2. Características del análisis clásico de sistemas

La principal limitación que presenta el enfoque clásico es la carencia de herramientas efectivas para llevar a cabo la tarea que le corresponde.

Las herramientas disponibles son abiertamente poco eficaces, además de ineficientes. Adolecen de deficiencias significativas de tal magnitud que no permiten conformar un producto -la especificación de requerimientos- claro y preciso en relación a lo que pretende describir, así como tampoco funcional en relación a la fase siguiente en el proceso de desarrollo del sistema, la fase de diseño, y menos aún fácil de entender.

Específicamente:

- El análisis clásico no provee de los elementos necesarios para conformar una especificación que refleje con efectividad las características del sistema desde el punto de vista de los datos y las transformaciones que los afectan. Por ende es difícil lograr descripciones adecuadas tanto de las características actuales como de la conformación que ha de tener tras el proceso de desarrollo al que se somete. Esto último es particularmente grave toda vez que los responsables operacionales del sistema sólo tendrá una visión exacta de aquel al momento de su operación normal. Antes de esta instancia, las concepciones que ellos tendrán serán meras aproximaciones muchas veces incompletas y otras tantas inexactas, debido justamente a la ineficacia de las comunicaciones producto de la inexistencia de modelos adecuados. Normalmente frente a la carencia de un modelo del sistema, se recurre a aquello que se tiene más a mano y esto corresponde obviamente al empleo de la narración escrita en lenguaje natural. Pero estas narraciones son en general poco precisas y normalmente se tornan voluminosas con lo cual dificultan aún más la comprensión del sistema. Además, en un texto narrado es particularmente difícil comprender las relaciones entre los elementos que conforman aquello que está siendo descrito, es decir, el sistema, debido a la pobre organización que como documento de especificación muestra el texto

narrativo. En estos casos, el analista debe ser consciente de las características propias de los lenguajes naturales. Ello con el agregado de ser además evidentemente tediosa de escribir y aún más de leer.

Uno de los instrumentos de que dispone el análisis clásico para describir un sistema como alternativa a las descripciones narrativas es el diagrama de flujo, instrumento de uso corriente en el diseño de un producto de software, sin embargo estos tienen el inconveniente de presionar al analista al uso de los símbolos normalizados para su confección, lo cual lleva a introducir elementos relativos a la implementación física del nuevo sistema, razón por la cual, en mayor o menor grado obliga a incluir en las tareas del análisis aspectos que deben abordarse en la siguiente fase de desarrollo de un sistema, la fase de diseño.

La fase de análisis debe centrar su interés en la descripción funcional del sistema enfatizando el "qué hacer" de la función sometida a tratamiento informático, sin especificar aún aquello que será automatizado en el sistema a implementar, ya que esto corresponde a la fase de diseño.

Chris Gane y Trish Sarson [GAN-87] citan al respecto a Fred Brooks quien señala que "el analista debe estar siempre preparado para mostrar una implementación para cualquier sección o aspecto que describa, pero no debe dictar la implementación" y lo expresa porque de ser así limita la libertad del diseñador.

4.2.1. Consecuencias del Análisis Clásico

Se puede decir que la consecuencia más notoria de la carencia de herramientas para enfrentar la tareas de análisis es la deficiente comunicación que se establece entre los especialistas y los responsables de la función sometida a tratamiento informático.

Esta deficiente comunicación se traduce en que muchas veces los conceptos e ideas que sobre el sistema tienen los analistas sesgan notoriamente la especificación. Por lo tanto el sistema generalmente no responde a lo que se esperaba de él, ya que por los problemas de comunicación los usuarios no logran hacer saber lo que necesitan y los analistas no alcanzan a comprender los requerimientos de los usuarios.

Otra consecuencia que también depende de lo anterior es que el sistema es ineficiente, tanto en su construcción como en su mantención.

Una manera de mejorar la comunicación usuario-analista es mediante la utilización de modelos, entendidos como una abstracción que representan la realidad que se pretende describir.

4.3. El Análisis Estructurado y el uso de Modelos

El oficio del arquitecto se puede considerar en alguna medida como análisis de la tarea del analista, ya que al igual que éste último, el objetivo específico del quehacer consiste en concebir y modelar para luego someter a un análisis específico y construcción aquello que se le requiere. El analista por su parte, que debe especificar para su diseño y construcción es un sistema informático.

El arquitecto usa modelos para la descripción de su quehacer. Específicamente emplea planos y maquetas, y con ellos se comunica con su cliente para mostrarle como será su casa antes de comenzar a construirla.

También usa aquellos para especificar al ingeniero o constructor lo que se quiere construir.

Son en consecuencia estos modelos básicamente un instrumento de comunicación.

La especificación es el producto principal de la fase de análisis y en ella debe describir el sistema sometido a desarrollo para posteriormente incorporarlo a la dinámica organizacional. En él se especifican las subfunciones que conforman el sistema y la forma en que se llevarán a cabo las tareas de cada una de dichas funciones, diferenciando entre automatizadas y manuales, describiendo además, las interfases que las relacionan.

En análisis es difícil describir funciones jerárquica y estructuralmente conformadas en un contexto organizacional complejo.

Obviamente es mejor usar gráficos que descripciones narrativas, ya que con ellos existe más certeza de generar respuestas favorables.

Los gráficos -o modelos de sistemas- son sin duda, además de efectivos como intentos de representación, instancias que mejoran indudablemente la comunicación entre el analista y el responsable de la función sometida a análisis. Por esta razón, el análisis estructurado plantea explícitamente el uso de modelos sugiriendo para el propósito específico de conformar una especificación de un sistema, los modelos del tipo redes.

Jay Forrester en su trabajo "Dinámica Industrial" [FOR-72] señala que los modelos han sido ampliamente aceptados como un medio para el estudio de los fenómenos complejos. Un modelo es un sustituto de algún equipo o sistema real. El valor de un modelo surge cuando éste mejora nuestra comprensión de las características del comportamiento, en forma más efectiva que si se observa el sistema real. Un modelo comparado con el sistema verdadero que representa, puede proporcionar información a costo más bajo y permitir el

conocimiento más rápido de las condiciones que no se observan en la realidad.

Una subdivisión importante de destacar dentro de los tipos de modelos es la que distingue entre modelos físicos y abstractos. Los primeros son los más fáciles de comprender, ya que generalmente corresponden a réplicas físicas de objetos en estudio. Los modelos abstractos están constituidos fundamentalmente por símbolos en donde el simbolismo puede ser un lenguaje gráfico, una representación estructural e incluso pensamiento.

Un caso particular que interesa destacar son los denominados grafos. En éstos, los elementos se representan por figuras geométricas, generalmente círculos, triángulos o puntos y se denominan nodos, componentes, etc. dependiendo de la convención establecida en cada caso particular.

Las interrelaciones entre los elementos se representan por líneas, flechas o curvas y se les denominan transiciones, conexiones, enlaces, etc.

Estos modelos tienen amplia aplicación, destacándose entre ellas su uso en la simulación, evaluación de rendimiento y descripción de sistemas entre otras.

En particular, cabe señalar que los grafos son instrumentos de suma utilidad cuando se requiere representar, analizar y/o explicar el comportamiento de un sistema de relativa o alta complejidad. El tipo de grafo más conocido lo constituyen las redes, por la capacidad que manifiestan éstos modelos para representar estructuras jerárquicas, compuestas por procesos e interfases. Se hace su empleo durante la fase de análisis en el desarrollo de un proyecto informático, ya que la realidad, el objeto de estudio, es una función o proceso compuesta por subprocesos relacionados mediante flujos de datos y con relaciones jerárquicas.

4. Ventajas de los modelos del tipo Redes

El uso de los modelos del tipo red en el desarrollo de proyectos informáticos tiene varias ventajas, entre las cuales cabe destacar las siguientes:

Es menos oneroso experimentar con él que hacerlo enfrentando los fenómenos reales.

Se pueden construir rápidamente.

Son fáciles de modificar.

Son fáciles de manejar.

Estos modelos son sin duda un instrumento eficaz para enfrentar la comunicación usuario-analista ya que:

- Permiten reflejar con efectividad la realidad o la concepción de un nuevo sistema.
- Permite comunicar de una forma más exacta y fiel las ideas que se tienen acerca de un sistema durante su desarrollo.
- Facilitan el refinamiento de la representación del sistema, toda vez que mediante ellos es posible, por su conformación modular y tras un proceso previo de evaluación de la realidad y reflejo, mejorar la calidad y aumentar la capacidad de comunicación del sistema implementado.

4.5. El Análisis Estructurado

Con el propósito de superar las limitaciones de lo que se ha venido a denominar "Análisis Clásico de Sistemas", han surgido nuevos planteamientos metodológicos, cada uno de los cuales provee un conjunto de herramientas para enfrentar las tareas fundamentales del análisis, en cuanto fase del desarrollo de un proyecto informático, es decir, el estudio de una cierta función, para reformularla en su conformación estructural y luego establecer los niveles de automatización que las subfunciones requieren a objeto de cumplir su cometido orgánico, con niveles adecuados de eficiencia y eficacia, requerimientos que posteriormente se someterán a diseño, construcción e implementación.

Las metodologías desarrolladas, por sus características propias, se pueden agrupar en tres categorías principales. La primera corresponde a aquella orientada a los flujos de datos, la segunda a las estructuras de datos y la última a los métodos basados en lenguaje.

Los métodos orientados a los flujos de datos centran su atención en el recorrido que siguen los datos desde que ingresan a la función sometida a desarrollo informático, desde una función relacionada, hasta que se dirigen a otra función relacionada, identificando todas las instancias en las cuales aquellos sufren alguna transformación.

Estos métodos han sido descritos principalmente en los trabajos de Tom de Marco [DEM-79] y Chris Gane & Trish Sarson [GAN-87].

Los métodos orientados a las estructuras de datos identifican, como lo señala Roger Pressman [PRE-88], los elementos de información y las acciones o procesos y los modelan de acuerdo a la jerarquía o estructuras de información del problema.

Estos fueron desarrollados principalmente por Kennett Orr [ORR-77], en base a trabajos anteriores de Jean D. Warnier [WAR-74] y se les conoce normalmente como Desarrollo de sistemas estructurados de datos.

Warnier crea al respecto una notación especial para la representación de los datos en una perspectiva jerárquica, utilizando las tres instrucciones básicas características de la programación estructurada como lo son la secuencia, selección e iteración, y comprueba que la estructura del software puede obtenerse directamente de la estructura de datos. Orr amplía la utilización de los instrumentos que presenta Warnier ahora para su empleo en la descripción de la función organizacional objeto de análisis.

M. Jackson [JAC-83], a su vez, postula el denominado "Desarrollo de sistemas de Jackson" y se centra en la identificación primero de las entidades, o sea, en los objetivos o funciones que necesita el sistema para producir o usar información y segundo en las acciones, es decir, en los sucesos que ocurren en el mundo real que afectan a las entidades.

Las acciones que afectan a cada entidad se ordenan temporalmente y se representan según ciertos diagramas también sugeridos por Jackson.

Luego las entidades y las acciones se representan como un modelo del proceso, definiéndose las relaciones entre el modelo y el mundo real.

Los métodos basados en lenguaje, como también señala R. Pressman [PRE-88], utilizan un lenguaje de especificación formal para modelar el sistema, facilitando el procesamiento automatizado para descubrir inconsistencias, omisiones y otros errores.

Estas herramientas pueden clasificarse desde distintos puntos de vista. Algunas se orientan a la automatización de la generación y mantenimiento en un determinado método, concebido originalmente para desarrollarse de manera manual. Estas producen diagramas, proveen de métodos para el particionamiento del problema, mantienen una jerarquía de datos en relación al sistema, proporcionan heurísticas o criterios para detectar inconsistencias en la especificación y entregan elementos para actualizar los datos e incorporar las relaciones entre los elementos que surgen en una nueva representación del sistema.

Otras herramientas proveen de una notación especial, generalmente un lenguaje de especificación de requerimientos, que puede ser procesado automáticamente. En estos casos los requerimientos se describen mediante un lenguaje de especificación, que combina indicadores de palabras claves con explicaciones en lenguaje natural. De este modo, el lenguaje de especificación se envía a un procesador que genera una especificación de requerimientos y un diagnóstico acerca de la consistencia y organización de la especificación.

Entre estas herramientas se destacan SADT [ROS-77], que significa System Analysis and Design Technique, SREM que quiere decir Software Requirement

Engineering Methodology [ALF-85], PSL/PSA que significa Problem Statement Language/Problem Statement Analyzer [TEI-77], y otros.

En este contexto de metodologías para enfrentar las tareas correspondientes a la fase de análisis, se encuentra el análisis estructurado.

El Análisis Estructurado es una metodología comprendida dentro de los métodos orientados a los flujos de datos que consisten, específicamente, en someter a estudio una cierta función para reconfigurarla estructuralmente, y luego establecer el grado de participación del procesamiento automatizado en uno o más de los subprocesos que conforman su estructura. Provee, en consecuencia, los elementos necesarios para el modelado de los elementos -los subprocesos y los flujos de datos- de la estructura de un sistema sometido a tratamiento informático, tanto para la representación de la estructura en su conformación actual como para la representación de la configuración obtenida tras el proceso de análisis.

Proporciona también los elementos indispensables para describir la composición de la interfase, incluyendo los almacenamientos de datos, entendidos como dispositivos temporales, y aquellos necesarios para formular los procedimientos que deben regir las transformaciones que cada subproceso del último nivel de la jerarquía estructural debe llevar a cabo, cuando recibe un flujo de datos de entrada para convertirlo en otro flujo de datos ahora de salida y con destino a otros subprocesos.

Dado el carácter esencialmente jerárquico de toda unidad organizacional, el análisis estructurado provee, en consecuencia, de los recursos indispensables para modelar y representar particionada y jerárquicamente la función objeto de estudio. Este modelo jerárquico -un modelo del tipo grafos- se denomina diagrama de flujo de datos, y con él se pretende describir la estructura de la función en términos de redes de procesos conectados por flujos de datos.

Del mismo modo, y con similares propósitos, entrega los instrumentos para la representación de las interfases de la estructura, es decir, de los flujos de datos que conectan los procesos. Estos, en el análisis estructurado se definen en un diccionario de datos, en el cual, mediante ciertos elementos notacionales se puede describir el contenido y las relaciones internas de cada uno de los flujos de datos existentes en la estructura funcional.

Para la descripción de los procedimientos, el análisis estructurado provee de un conjunto de elementos para explicitarlos exhaustivamente, sin redundancias y sin ambigüedades. Estos quedan comprendidos en un subconjunto del lenguaje denominado español estructurado.

Por último, para la descripción de las operaciones de acceso a los almacenamientos de datos, proporciona los denominados diagramas de acceso inmediatos que destacan y priorizan las diferentes claves de acceso a los archivos, exigidos normalmente por motivos de eficiencia operacional en los procesos.

El uso de estos instrumentos en los términos señalados permiten conformar un documento único denominado especificación estructurada, en el cual se plasma el resultado del trabajo realizado en la etapa de análisis y que es aquel que sirve de base para la fase siguiente en el desarrollo de un proyecto informático.

Por lo tanto, la especificación estructurada, producto principal de la fase de análisis, consta de cuatro componentes básicos:

- Diagrama de Flujo de Datos.
- Descripción de Procesos.
- Diccionario de Datos.
- Diagrama de Acceso Inmediato.

Estos elementos en conjunto, permiten conformar un documento mediante el cual se puede describir cualquier sistema. Se les considera como las herramientas básicas del Análisis Estructurado.

Referencias

- ALF-85, Alford, M., SREM at the age of eight; The Distributed Computing Design Systems, Computer, vol 18, No.4, April 1985.
- DEM-79, De Marco, T., Structured Analysis and System Specification, Yourdon Press, New York, 1979.
- DIJ-68, Dijkstra, E., Go To Statement Considered Harmful, Communications of the ACM, vol. 11, No. 3, March 1968.
- FOR-72, Forrester, J., Dinámica Industrial, El Ateneo, Buenos Aires, 1972.
- GAN-87, Gane, C., T. Sarson, Análisis Estructurado de Sistemas, El Ateneo, Buenos Aires, 1987.
- JAC-83, Jackson, M., System Development, Prentice Hall, 1983.
- BOH-62, Bohm, B. W., G. Jacopini, Nuove tecniche di programmazione semplificanti la sintesi di macchine universali di Turing. Rend. Acc. Naz. Lincei, Vol. 8, No. 32 (June 1962), pp. 913-922.
- ORR-77, Orr, K., Structured System Development, Yourdon Press, New York, 1977.

- PRE-88, Pressman, R., Ingeniería de Software un enfoque práctico, Mc Graw Hill, Madrid, 1988.
- ROS-77, Ross, D., K. Schoman, Structured Analysis for Requirements Definition, IEEE Trans. Software Engineering, vol. 3, No. 1, January 1977.
- ROS-85, Ross, D., Applications and Extensions of SADT, Computer, vol. 18, No. 4, April 1985.
- TEI-77, Teichroew, D., E. Hershey, PSU/PSA: A computer aided technique for structured documentation and analysis of information processing systems, IEEE Trans. Software Engineering, Vol. 3, No. 1, 1977, pp. 41-48.
- WAR-74, Warnier, J. D., Logical Construction of programs, Van Nostrand Reinhold, 1981.
- YOU-78, Yourdon, E., L. Constantine, Structured Design, Yourdon Press, 1978.

El Modelado del Sistema

El análisis estructurado se basa fundamentalmente en el uso de representaciones gráficas, modelos, para reflejar el sistema sometido a desarrollo informático. No cabe duda que una de las formas de descripción más usadas, y seguramente una de las más eficaces, es aquella lograda mediante representaciones gráficas.

Los modelos que se utilizan en el análisis estructurado son del tipo redes o grafos, y mediante ellos se pretende representar los flujos de datos que circulan por el sistema y los procesos que los afectan y transforman.

La representación gráfica de sistemas es, en esencia, una forma de utilización de la teoría de grafos* para la resolución de problemas específicos [ACE-88].

*Un grafo G , es un par (N,A) en el que:

N es un conjunto finito, no vacío de objetos denominados nodos (o vértices)

$A \subseteq N \times N$ es un conjunto de pares de nodos, denominados arcos.

Existe un arco que une N_i a N_j si y solo si $(N_i, N_j) \in A$. Un nodo se representa gráficamente mediante un punto \circ o una circunferencia (\circ) y un arco por una flecha (\rightarrow) .

de un fin determinado. Este tipo de diagrama se explica por la red configurada por los niveles 1.1, 1.2 y 1.3. Lo esencial reside en identificar y reflejar aquéllos elementos y sus interrelaciones.

Los elementos en un grafo son representados por medio de figuras geométricas, generalmente círculos, cuadrados o puntos y se denominan nodos, componentes o vértices, dependiendo de la convención establecida en cada caso particular. Por cierto que estos elementos pueden ser conceptuales, objetos o sujetos. En todo caso, cabe destacar que los elementos a considerar deben ser sólo aquéllos sin los cuales el sistema no puede representarse exhaustivamente.

En un grafo, las interrelaciones entre los elementos se representan por líneas con punta de flecha o arcos que los conectan entre sí y se les denominan transiciones, conexiones, o enlaces.

Estos grafos son los que se utilizan para modelar el comportamiento de los datos al interior del sistema, modelos generalmente conocidos como diagramas de flujo de datos. Cada sistema queda descrito por un conjunto de diagramas derivados de la conformación jerárquica del sistema, más específicamente, de la función o proceso sometido a tratamiento informático. Estos diagramas se utilizan para representar el sistema, en todas las fases de la etapa de análisis en que son requeridos, es decir, cuando se describe el sistema como funciona actualmente y cuando se propone un nuevo sistema redefinido estructuralmente. Lo esencial de cada diagrama es que explicita las relaciones de las subfunciones derivadas en términos de los flujos de datos que las ligam.

Los diagramas de flujo de datos, por lo tanto, consisten en una red jerárquica que muestra el movimiento de los datos y sus transformaciones a través del sistema, en donde los nodos son las transformaciones y los arcos los flujos de datos.

Estos diagramas son jerárquico, puesto que cualquier nivel de representación se explica por la red que relaciona los elementos del nivel inmediatamente inferior, tal como lo muestra la figura 5.1.

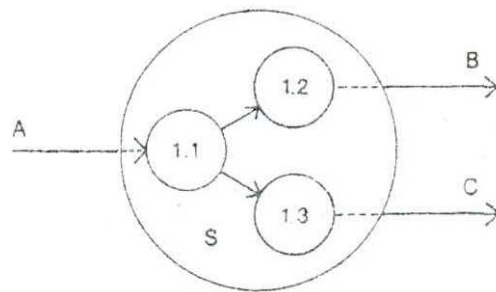


Figura 5.1: Diagrama Jerárquico

En un lenguaje coloquial, se acostumbra llamar a estos "diagramas de burbujas".

Cualquier nivel de la estructura jerárquica del proceso o función, el diagrama de flujo de datos, muestra flujos de datos y los procesos a los que aquellos se refieren, tal como aparece el ejemplo de la figura 5.2. Este se refiere a una función, obtenida por desagregación de una función representada de manera más global en los niveles superiores del diagrama de flujo de datos. En ella, un cierto cliente de una casa comercial solicita un crédito, el cual debe ser aceptado o rechazado. Si el crédito es aceptado, se debe facturar la operación comercial entregando el original de la factura al cliente y destinando una copia de ella al proceso de mantención de cuentas corrientes de deudores.

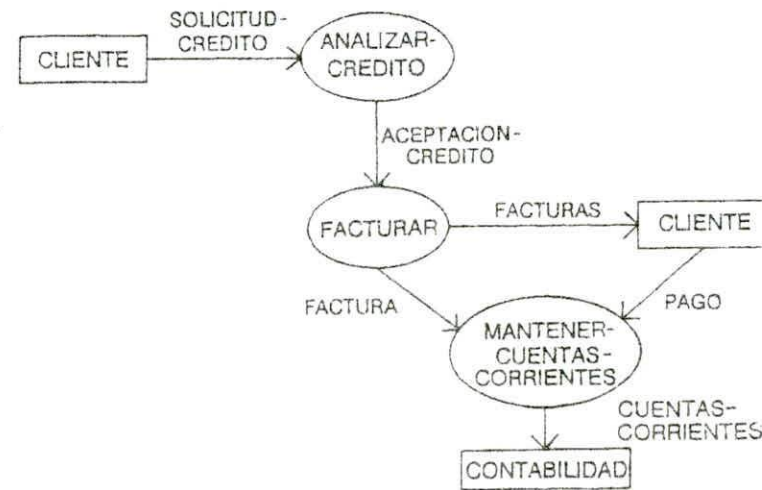


Figura 5.2: Ejemplo de un D.F.D.

5.1. Características de los diagramas de flujo de datos

Los diagramas de flujos de datos poseen varias características las cuales les permiten convertirse en la herramienta fundamental del análisis estructurado, toda vez que se constituyen en un instrumento eficaz para el modelado del sistema que se pretende representar.

Entre aquellas características, cabe destacar:

- Los diagramas de flujos de datos son gráficos, lo que los hace más explícitos y comprensibles, tanto para el usuario, como para el analista.
- Los diagramas de flujos de datos son particionados, es decir, representan el sistema a diferentes niveles de complejidad, desde lo más global a lo más detallado. Para ello la función o proceso objeto de análisis se desagrega sucesivamente en subfunciones. Este proceso de refinamiento sucesivo termina cuando las subfunciones derivadas en un cierto nivel ya no se pueden subdividir, porque al interior de ellas no se generan flujos de datos explícitos, razón por la cual estos se describen en términos de los procedimientos mediante los cuales es posible obtener los flujos de datos de salida, que se generan a partir de los flujos de datos de entrada que reciben.

Estos procedimientos corresponden, en consecuencia, a cada una de las burbujas del último nivel en el diagrama. Los diagramas de flujo de datos son por lo tanto multidimensionales.

- Los diagramas de flujos de datos tienen como componentes fundamentales los flujos de datos, y las transformaciones que los afectan al interior del proceso o función sometido a análisis. Esto significa que lo que en ellos se explicita es el recorrido que siguen los datos desde una función relacionada de entrada hasta una función relacionada de salida.
- Los diagramas de flujo de datos no reflejan la dinamicidad y la temporalidad del sistema que se pretende representar. Sólo se especifican los flujos de datos y los procesos que los transforman.





En ellos, en consecuencia, no se consideran las instancias de control, que determinan la oportunidad, periodicidad y condiciones que han de regir las transformaciones a que se someten aquellos flujos de datos.

- Utiliza un número restringido de símbolos consistentes para la representación del sistema. Los diagramas de flujo sólo utilizan cuatro símbolos diferentes, para representar funciones, funciones relacionadas, flujos de datos y archivos. Su uso es consistente puesto que cada uno de ellos se emplea sólo de una manera, aumentando con ello la capacidad de comunicación de la especificación.

- Los diagramas de flujo de datos son fáciles de mantener puesto que cualquier modificación que se suscita en la función, se puede absorber con mínimo esfuerzo por el carácter modular y jerárquico que poseen.

2. Elementos de los Diagramas de Flujos de Datos

Los diagramas de flujos de datos, permiten el modelado de un sistema en términos de cuatro elementos básicos:

- Los flujos de datos 
- Los procesos o funciones 
- Los archivos 
- Los orígenes o destinos 

Cada uno de estos elementos se representa mediante un símbolo único. De este modo entonces:

- Los flujos de datos se representan por vectores, como un modo de reflejar el movimiento que tienen los datos dentro del sistema.
- Los procesos o funciones se representan por un círculo, o una burbuja y dan a entender que en ellos se realiza una transformación de ciertos datos de entrada en otros datos de salida.
- Los archivos se representan por una línea recta, significando con ello que se trata de una instancia de almacenamiento de datos.
- Los orígenes o destinos se representan por un cuadrado o caja, indicando con ello que son los generadores o receptores de los flujos de datos que entran o salen de la función o proceso sometido a tratamiento informático.

En la figura 5.3 se puede apreciar una porción de un diagrama de flujo de datos que incluye los cuatro elementos mencionados anteriormente.

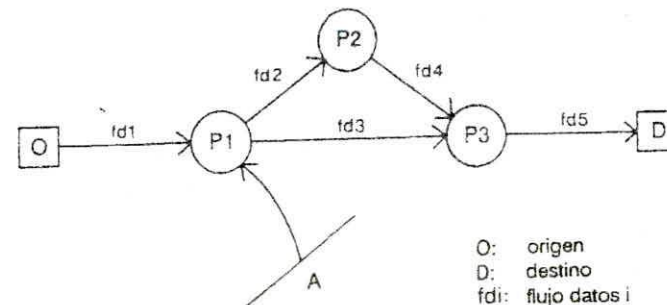


Figura 5.3: Elementos de un DFD.

Cada uno de los elementos de un diagrama de flujo de datos tiene sus propias particularidades y alcances.

5.2.1. Los Flujos de Datos

Los flujos de datos deben reflejar todos los movimientos que tienen los datos dentro del sistema, es decir entre procesos, como entrada o salida a los archivos, y también como entrada o salida a destinos y orígenes respectivamente.

Tom De Marco [DEM-79], señala en relación a ellos, que un flujo de datos es un conducto a través del cual fluyen paquetes de datos de composición conocida. Por paquete de datos se entiende un conjunto de datos, cuyos elementos fluyen lógicamente unidos y con un propósito común. Esto significa que fluyen juntos debido a los requerimientos funcionales del sistema y no por simple azar.

La composición del paquete de datos que fluye, queda especificada en el diccionario de datos, el cual corresponde a uno de los instrumentos con que se complementa los diagramas de flujos de datos.

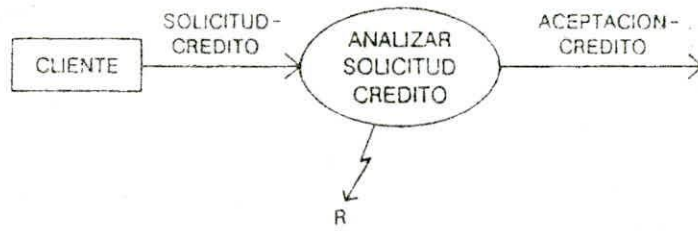


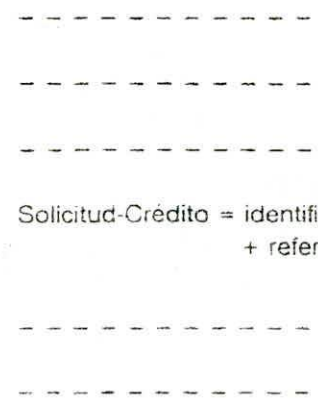
Figura 5.4: Ejemplo específico de DFD

La figura 5.4, muestra una subfunción, "Analizar Solicitud de Crédito" de la función ventas de una casa comercial que otorga créditos a sus clientes. Aquella subfunción debe estudiar una solicitud de crédito presentada por el cliente y rechazarla o aceptarla emitiendo en este último caso un documento de aceptación de crédito.

En este caso, "solicitud-crédito", es un flujo de datos, es decir, se trata de un paquete de datos cuya composición en caso de requerirse, se puede obtener del diccionario de datos del sistema.

La forma en que esta composición se explicita en el diccionario de datos, se aprecia en la figura 5.5.

DICCIONARIO DE DATOS



Solicitud-Crédito = identificación cliente + activo y pasivo + referencias bancarias + nombre aval.

Figura 5.5: Composición del flujo de datos.

Puede ocurrir que dos procesos estén conectados por dos o más flujos de datos. Si ello ocurre significa entonces que en este caso hay más de un paquete de datos involucrado en la relación entre los procesos, la figura 5.6 muestra dos procesos conectados por flujos de datos.

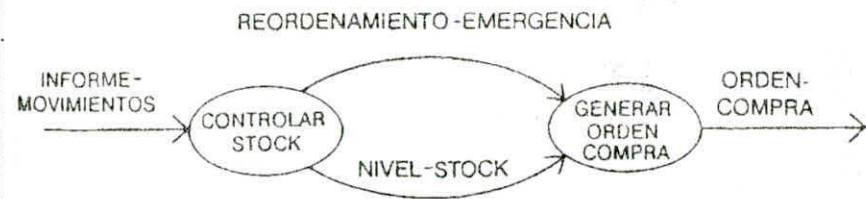


Figura 5.6: Relación entre procesos

El ejemplo de la figura 5.6 muestra una porción de un diagrama de flujo de datos correspondiente a un sistema de control de inventarios en el cuál la función compras emite una orden de compra toda vez que el nivel de stock alcanza cierto valor mínimo, valor que se verifica dos veces por semana. La orden de compra se emite cuando se alcanza un cierto nivel crítico fuera de los puntos de control semanales establecidos.

Si en este caso el flujo de datos "nivel-stock" fluye dos veces por semana y el flujo "reordenamiento-emergencia" fluye cada vez que se alcanza un nivel crítico

para un cierto ítem, entonces, con toda seguridad, ambos se producen por funciones distintas e incluso afectan procedimientos diferentes.

Por cierto que se puede dar el caso que ambos fluyan en un mismo momento, pero de ocurrir ello se trata de un evento fortuito ya que no está especificado que deba ocurrir así.

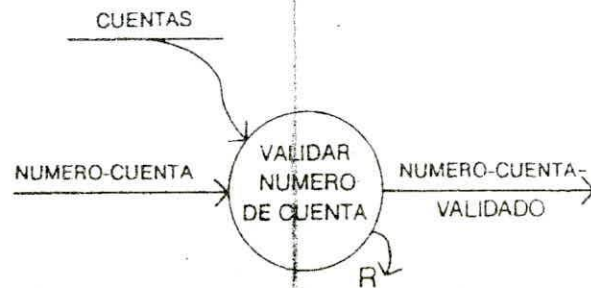
Cabe destacar que en un diagrama de flujo de datos no es recomendable representar como un solo flujo dos paquetes de datos, aunque operativamente se sometan al mismo trabajo y deban ser aplicados en un mismo momento, pues esto sólo entorpecería la comprensión del sistema, además de la dificultad que surgiría al tratar de encontrar un nombre adecuado que represente tal eventual flujo compuesto.

5.2.1.1. Convenciones respecto de los Flujos de Datos

En la mayoría de las unidades de informática, y específicamente en relación a los procedimientos que se establecen para la aplicación uniforme de la metodología, existen ciertas convenciones en torno a la forma de presentación de los diagramas de flujo de datos. Aunque estas pueden variar considerablemente de una realidad a otra, todas deben apuntar hacia la mayor eficacia y eficiencia de la representación en cuanto instrumento de descripción y comunicación.

Entre las normas de uso general, relativas a la representación de estos flujos se destacan:

- Escribir los nombres con mayúsculas y separar las palabras por guiones.
- Los nombres de los flujos de datos deben ser únicos para evitar confusiones.
- Los nombres utilizados deben ser representativos de los datos que contiene el flujo, es decir, deben dar a entender su contenido.



El ejemplo de la figura 5.7 muestra dos flujos de datos: "Número-Cuenta" y "Número-Cuenta-Validado".

El segundo flujo contiene obviamente más información que el primero, aunque ambos tienen igual composición.

Si "número-cuenta" es 0141-7, por cierto que "número-cuenta- validado" será también 0141-7. Pero aunque físicamente son iguales, hay una diferencia sustantiva, en el segundo caso se trata de un número de cuenta que ya ha sido validado.

- Para el caso de flujos convergentes o divergentes se sugiere utilizar bifurcaciones y sólo un nombre para representarlos, tal como lo muestra la figura 5.8.
- No es necesario identificar los flujos que van o vienen de un archivo, pues el nombre de éste indica qué es aquello que contiene y, por ende, lo que fluye de él.

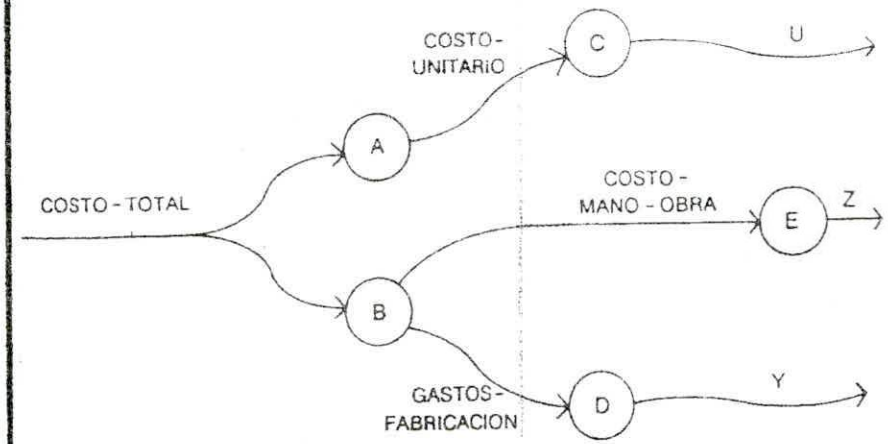
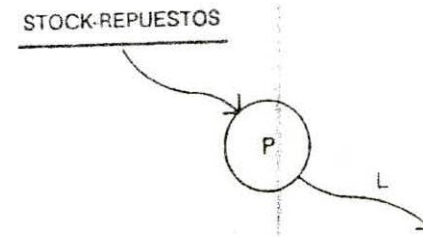


Figura 5.8: Flujo de datos convergentes y divergentes



La figura 5.9, muestra un proceso P que retira datos de un archivo "stock-puestos" para llevar a cabo su transformación. De la lectura del diagrama puede deducir que lo que fluye del archivo al proceso son datos referidos al ítem que componen el archivo.

Si se desea un mayor detalle acerca del contenido del flujo -o del archivo-, decir, conocer su composición, se debe recurrir al diccionario de datos.

5.2.1.2. Revisión de los flujos de datos

Entre los errores más comunes que se cometen al dibujar un diagrama de flujos de datos, está el incluir en ellos flujos mandatorios. Los flujos mandatorios son aquellos que no tienen datos como contenido, pues corresponden más bien a indicadores o activadores de procesos.

Los flujos mandatorios, no deben ser incluidos en los diagramas de flujos de datos, sino más bien, en las especificaciones de procedimientos, por cuanto al ser indicadores o activadores de procesos se relacionan con las reglas operacionales que deben regir las transformaciones.

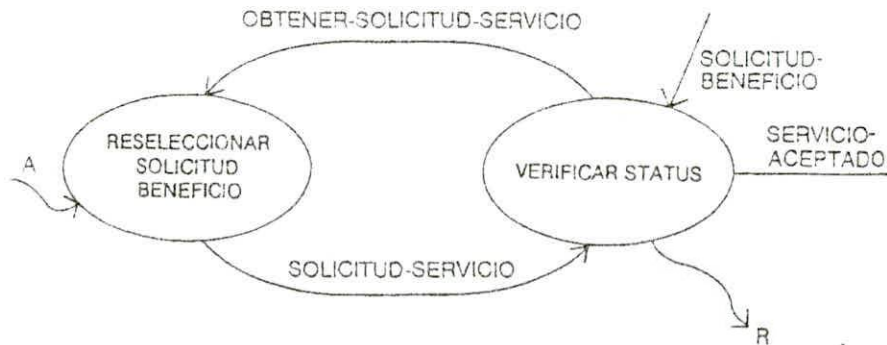


Figura 5.10: Flujo "mandatorio"

En la figura 5.10, se aprecia un fragmento de un diagrama de flujo de datos, referido a la secretaría de un club social, el cual otorga ciertos beneficios a los socios que tienen al día sus cuotas. Para otorgarles dichos beneficios, primero se verifica el status del socio, para lo cual, éste debe llenar un documento con sus datos personales.

En este caso, tal como se puede apreciar en la figura 5.10, el flujo "obtener-solicitud-servicio" no tiene datos y dado su carácter es más bien una instancia que corresponde a los procedimientos que rigen el proceso de verificación-de-status. Estos son los llamados flujos mandatorios y justamente por ello no deben incluirse en los diagramas de flujos de datos.

2.2. Los Procesos

Los procesos corresponden a las componentes estructurales de la función que llevan a cabo las transformaciones que sufren los flujos de datos, por lo tanto, desde el punto de vista de los datos, existe un proceso cada vez que un flujo de datos sufre una transformación determinada.

En otras palabras, un proceso es una transformación del tipo entrada/salida, tal como lo muestra la figura 5.11.

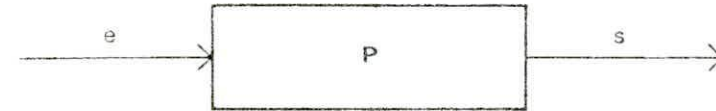


Figura 5.11: El proceso de transformación.

Por ejemplo, una subfunción de la función relaciones industriales de una empresa productiva debe seleccionar un candidato que reúna ciertos requisitos básicos:

- edad entre 20 y 30 años
- escolaridad: egresado escuela industrial
- sexo: masculino.

Todos los postulantes llenan una solicitud y la entregan en el Departamento de Relaciones Industriales de la empresa. Una unidad encargada de la selección realiza el chequeo de los antecedentes básicos rechazando aquellas que no los cumplen y enviando las que sí los cumplen a una segunda fase de selección. Esta transformación queda reflejada en el fragmento de diagrama que muestra la figura 5.12.



Figura 5.12: Ejemplo de proceso

Los procesos, en cuanto transformaciones del tipo entrada- salida, se representan por burbujas y deben llevar un nombre representativo y descriptivo, junto a un número correlativo dentro de la función.

Debido a que los diagramas de flujos de datos son jerárquicos, como lo muestra la figura 5.13, además de su correlatividad, el número de la burbuja debe reflejar el nivel de desagregación correspondiente. Para esta numeración jerárquica por niveles se utiliza como notación el punto.

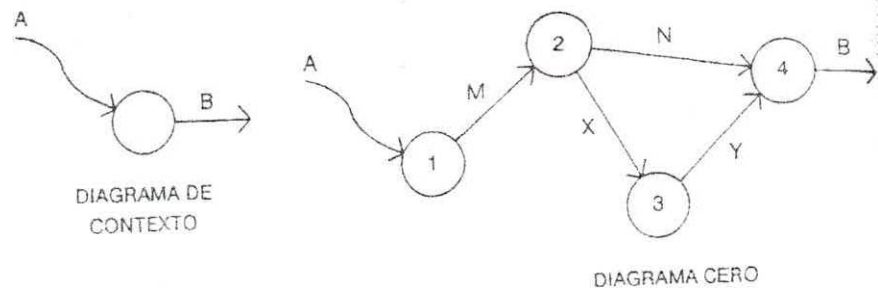


Figura 5.13: Jerarquía y numeración de los DFD

La figura 5.13 muestra un sistema a tres niveles. El diagrama 3 corresponde a un tercer nivel jerárquico y a un segundo nivel de desagregación.

5.2.3. Archivos

Para los propósitos del análisis de sistemas un archivo es un depósito temporal de datos. Estos pueden estar contenidos en un disco, un diskette, un cassette, o una carpeta y en sistemas automatizados pueden corresponder a archivos tradicionales o a bases de datos.

Los archivos en un diagrama de flujo de datos se representan por una línea recta, o por dos líneas cuando involucra las actividades de dos procesos, como muestra en la figura 5.14.

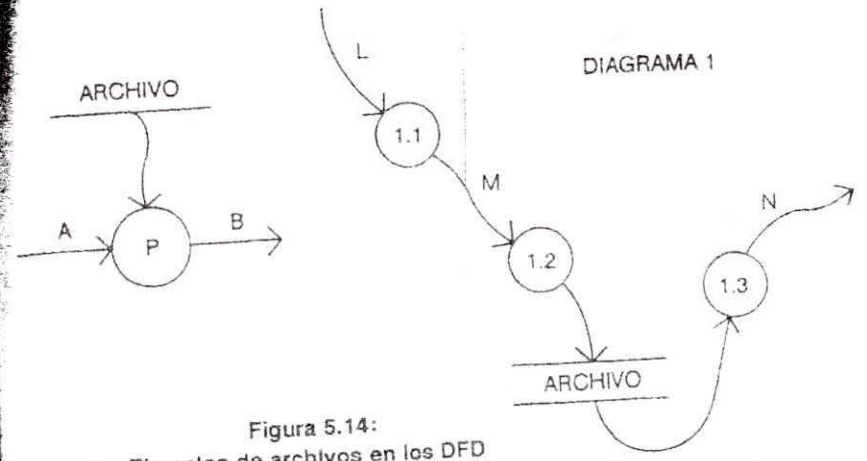


Figura 5.14: Ejemplos de archivos en los DFD

En el diagrama de flujo de datos, se especifican por su nombre y su contenido, se detalla en el diccionario de datos.

Como regla general se ha establecido que no se especifiquen por códigos, ya que los diagramas de flujo de datos deben explicarse a sí mismos, es decir, deben tender a la autosuficiencia, y ello obviamente se ve menoscabado cuando se recurre a la utilización de siglas o códigos. Un aspecto que cabe destacar en la diagramación del modelo es la dirección de los flujos que unen archivos con procesos.

Por ejemplo, si se desea verificar la corrección ortográfica de un conjunto de palabras escritas y para ello se recurre al diccionario de la lengua española, entonces éste último será un archivo, el cuál es preciso consultar para realizar la tarea correspondiente al proceso. En este caso, obviamente la dirección del flujo, especificada por una punta de flecha, necesariamente debe apuntar sólo al proceso, ya que el movimiento de los datos, las palabras, sólo puede fluir en esa dirección, puesto que no es posible incorporar palabras al diccionario a partir de un proceso como "verificar corrección ortográfica", tal como lo muestra la figura 5.15.



Figura 5.15: Ejemplo de lectura de un archivo

En la figura 5.16 el flujo representa una situación en la cual los datos fluyen desde el archivo al proceso, y del proceso al archivo. Para este caso se utiliza una doble flecha en el flujo que une un archivo y un proceso.

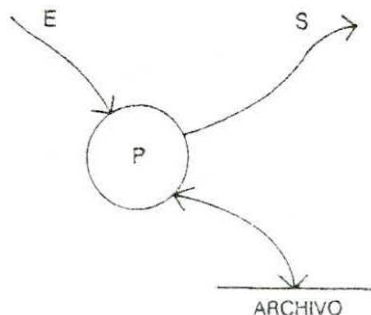


Figura 5.16: Ejemplo acceso "doble" a un archivo

En este caso, la porción del diagrama corresponde a una situación en la cual el archivo recibe y envía datos desde y hacia el proceso.

Se puede señalar, como una regla general, que los flujos que relacionan archivos y procesos deben sólo mostrar el flujo neto de datos desde y hacia el archivo.



Figura 5.17: Ejemplo de flujo neto en los archivos

Figura 5.17, muestra un proceso que actualiza un archivo de ítemes de repuestos, y que además emite un documento con los nuevos valores de los ítemes.

En este caso podría pensarse en utilizar la doble flecha si se argumenta que no es posible actualizar un archivo sin leerlo primeramente. Pero dado que la indicación lectura no involucra flujo de datos, la flecha sólo se direcciona hacia el archivo, ya que es ese el sentido en que queda reflejado el flujo neto de datos.

En cualquier sistema no puede existir un archivo que sólo tenga asociados flujos de datos de entrada o flujos de datos de salida.

Si así sucede en algún diagrama equivalente a cierta subfunción, derivada de

una función más global, lo que debe ocurrir, necesariamente, es que en otro diagrama del mismo nivel de desagregación aparezca el mismo archivo con la entrada o salida de la cual carece en el diagrama anterior, tal como lo muestra la figura 5.18.

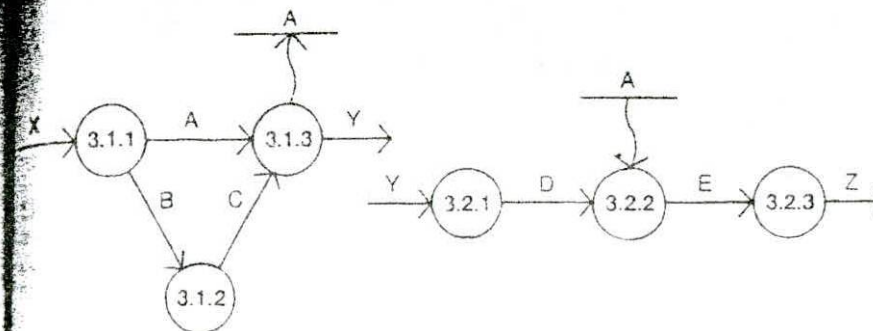


Diagrama 3.1

Diagrama 3.2

Figura 5.18: Utilización de los archivos en más de un DFD

En los diagramas de flujo de datos, los archivos que utilizan las diferentes subfunciones deben aparecer explícitamente. Existen archivos que se ubican en la interfase entre dos procesos y otros que sirven de soporte solamente a una subfunción. En este último caso, aunque el archivo obviamente es interno a él, se incluye en el diagrama debido a la necesidad de destacarlo frente a la posibilidad que existe de automatizar su manejo, sobre todo cuando su volumen es significativo y cuando se precisa mejorar los métodos de acceso.

5.2.4. Los orígenes y destinos

Los orígenes y destinos son los generadores o receptores de los flujos de datos que llegan o se van de la función sometida a análisis. En otras palabras, un origen o destino es una función o proceso, fuera del contexto del sistema, pero que se relaciona con él en la medida en que se constituye en un generador o receptor de los datos del sistema.

Los orígenes y destinos se representan por cuadrados y en lo específico determinan el ámbito de la función, tal como lo muestra el ejemplo de la figura 5.19.

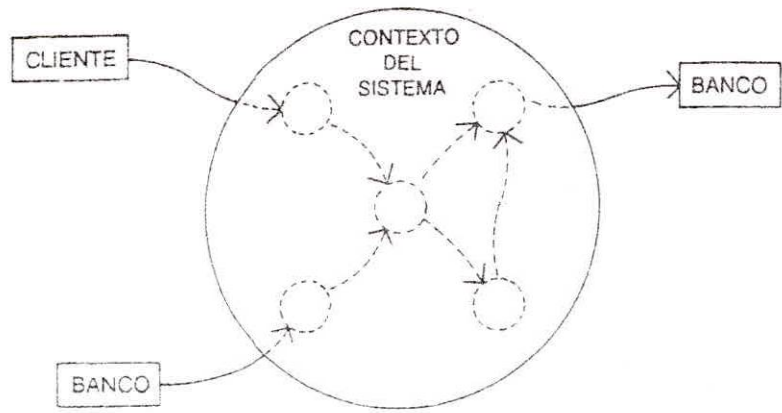


Figura 5.19: Orígenes y destinos de una función

Cabe destacar que por el carácter jerárquico de los diagramas de flujos de datos, los orígenes y destinos sólo se explicitan en el nivel más global de la representación.

5.3. Los Diagramas de Flujo de Datos y la Descripción de Procedimientos

La figura 5.20 muestra un diagrama de flujo de datos que incluye cierta información relativa a procedimientos. Específicamente, ésta expresa que los dos flujos de datos de entrada deben estar presentes para que el proceso pueda realizar su trabajo y además que se obtendrá sólo uno de los dos flujos de datos de salida. Por lo tanto:

- * significa Y, conjunción de flujo de datos.
- + significa O, es decir, disyunción de flujo de datos.

No obstante el mayor nivel de entendimiento que proveen estas notaciones, no es conveniente incluir aspectos relativos a procedimientos en los diagramas de flujo de datos, salvo en situaciones como la que muestra la figura 5.20, y con la expresa pretensión de facilitar su lectura. Agregar otros elementos, más que ayudar, disminuye su efectividad como intento de comunicación.

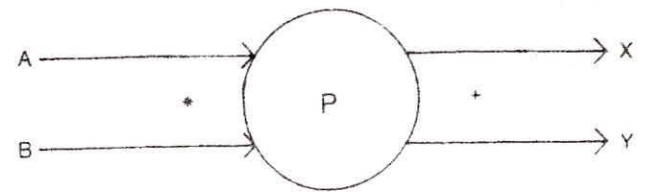


Figura 5.20: Notaciones adicionales al DFD

Si la porción de diagrama de flujo de datos de la figura 5.20 no contiene notación a nivel de procedimientos, entonces de la lectura de dicho diagrama sólo es posible deducir que hay dos flujos de datos que van hacia el proceso y dos que salen, sin especificar algo acerca de como se interrelacionan. Esta situación aparentemente limita la capacidad de comunicación efectiva del sistema. Sin embargo, dado que los diagramas de flujo de datos sólo pretenden presentar el particionamiento del sistema, y éste depende de la complejidad de las interfaces, no es inadecuado explicitar con posterioridad la especificación de los procedimientos, sobre todo si el particionamiento se puede establecer siempre a partir de los flujos y procesos.

Obviamente para la comprensión cabal de una función, desde el punto de vista de los datos que por ella circulan y maneja, no basta con el modelo que representa los flujos de aquellos datos y las transformaciones llevadas a cabo por los subprocessos que la conforman. Es preciso especificar los procedimientos en virtud de los cuales se llevan a cabo cada una de las transformaciones que al interior de la función afectan a los flujos.

El planteamiento metodológico del análisis estructurado sugiere que la especificación de los procedimientos, que han de regir las transformaciones que debe llevar a cabo cada proceso de último nivel, se incluya en un documento complementario a los diagramas de flujo de datos, denominado justamente especificación de procedimientos. Por lo tanto, el incluir en los diagramas de flujo de datos notaciones asociadas a las reglas que han de regir la operaciones sólo aumenta la redundancia.

Un diagrama de flujo de datos, ni siquiera debe mostrar las relaciones entre los flujos de datos de entrada y salida. Tom De Marco [DEM-79] cita al respecto a Jerry Weinberg, quien en "Introduction to General System Thinking", habla de la ley de Lump. Esta expresa que "Si queremos aprender algo no debemos tratar de aprenderlo todo, al menos no al comienzo".

Es, por lo tanto, normal que la lectura de un diagrama de flujo de datos haga necesario recurrir a la descripción de procesos para lograr una comprensión total del sistema, como ocurre por ejemplo en la figura 5.21, en la cual no es posible saber con exactitud cual es el destino de los flujos A y B. Allí no se puede establecer si el flujo A se transforma en C o D, o si el flujo B se transforma

en C o D. La forma en que se establecen las relaciones las explicita la especificación de procedimientos.

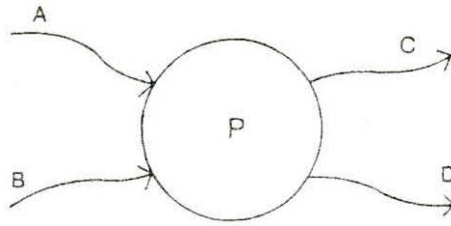


Figura 5.21: Un proceso con varios flujos

5.4. Pautas para el Diseño de un Diagrama de Flujos de Datos

Durante la etapa de análisis en el desarrollo de un sistema, existen dos instancias en las cuales es preciso configurar un diagrama de flujo de datos: cuando se describe el sistema tal como funciona actualmente, y cuando es preciso proponer un nuevo sistema. Estas dos instancias se traducen en un diagrama de flujo de datos físico actual, un diagrama de flujo de datos lógico actual, un diagrama de flujo de datos lógico nuevo y un diagrama de flujo de datos físico nuevo.

El primer uso que se hace de los diagramas de flujo de datos, es su empleo como una herramienta que permite documentar el sistema tal como funciona actualmente. En estos diagramas se explicita lo que realmente éste hace, sin considerar en absoluto en esta instancia aquello que debe hacer. En virtud de ello, en este primer momento de la tarea de análisis, se debe reflejar en el diagrama de flujo de datos lo que el especialista -el analista- ha observado directamente, y lo que los involucrados en las tareas habituales de la función o proceso han expresado y comunicado en relación a la forma en que se llevan a cabo las tareas de la función, desde la perspectiva de los datos y la información que circula por aquellas funciones o procesos.

Existen, sin duda, diferentes pautas de acción para llevar a cabo la confección de un diagrama de flujo de datos cuando se pretende reflejar la forma en que un sistema desarrolla actualmente su quehacer organizacional. Sin embargo, existen ciertas tareas que no es posible obviar tanto en términos de oportunidad como en contenido.

Estas acciones son:

- 1.- Definir de las fronteras del sistema.
- 2.- Identificar e incorporar los flujos de datos de entrada y salida.
- 3.- Identificar las subfunciones y flujos de datos intermedios.
- 4.- Asignar nombres a los flujos.
- 5.- Asignar nombres a los procesos.
- 6.- Incorporar archivos.
- 7.- Verificar el modelo conformado.

5.4.1. Definición de las fronteras del sistema

La decisión de determinar el contexto de la función o proceso sometida a análisis, es una decisión que debe tomarse al comienzo de la fase de análisis, a menudo con muy pocas consideraciones, principalmente aquellas contenidas en la definición del problema y en el estudio de factibilidad, las dos etapas que preceden al análisis en la metodología de desarrollo de un sistema informático. En muchas ocasiones determinar el contexto no es una tarea obvia y más aún es una cuestión de juicio y sentido.

Al determinar el contexto, es preciso tener en consideración que aquél debe ser lo suficientemente amplio como para incluir todos los aspectos funcionales relevantes para el desarrollo del sistema, pero también lo suficientemente pequeño como para no incluir aquellos irrelevantes.

Del mismo modo, es preciso tener en cuenta que cualquier aspecto no considerado, será ignorado definitivamente durante todo el desarrollo del sistema, salvo que durante este proceso se opte por redefinirlo como proyecto.

Resulta obvio entonces, que frente a una situación de duda, lo más aconsejable es incluir, más que descartar.

Determinar el contexto es, en consecuencia, establecer las fronteras del sistema, es decir, distinguir entre la función objeto de análisis y las funciones relacionadas tanto de entrada como de salida, las cuales conformarán el entorno del sistema.

5.4.2. Identificación e Incorporación de los flujos de datos de entrada y salida

Una vez establecidos entonces los límites del sistema, es preciso especificar las entradas y salidas, las cuales corresponden, en los diagramas de flujos de

datos, a flujos de datos que llegan y salen del sistema respectivamente, y asumirán como tales cuando sea posible considerarlos como paquetes de datos, es decir como una unidad que circula y que es procesada como un todo.

Las entradas tienen su origen en las funciones relacionadas de entrada, y las salidas, como destino, las funciones relacionadas de salida.

Cabe destacar que generalmente los involucrados en el quehacer de la función o proceso identifican éstos flujos denominándolos de manera específica y representativa.

Establecidas entonces las fronteras del sistema, e identificadas las entradas y salidas, se puede comenzar a conformar el modelo del sistema, para lo cual se sugiere representarlos como flujos de datos, en el contorno de una burbuja que representa al sistema como un todo, tal como lo muestra la figura 5.22..

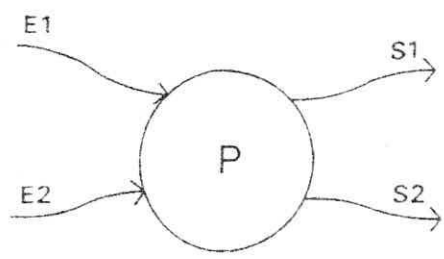


Figura 5.22: El sistema como un todo

5.4.3. Identificación de las subfunciones y flujos de datos intermedios

La descripción del sistema a partir de los flujos de datos de entradas y salida identificados, continúa con el seguimiento de los flujos de datos de entrada a través de la función o proceso. Se deben separar aquellos flujos de datos que el usuario ve como un todo, pero que en realidad son un conjunto de flujos independientes.

Los flujos que circulan al interior del diagrama deben estar obviamente conectados con los flujos de la periferia, y en el seguimiento es preciso introducir burbujas cuando un flujo de datos o un conjunto de flujos de datos es transformado para generar otro u otros flujos, tal como lo muestra la figura 5.23.

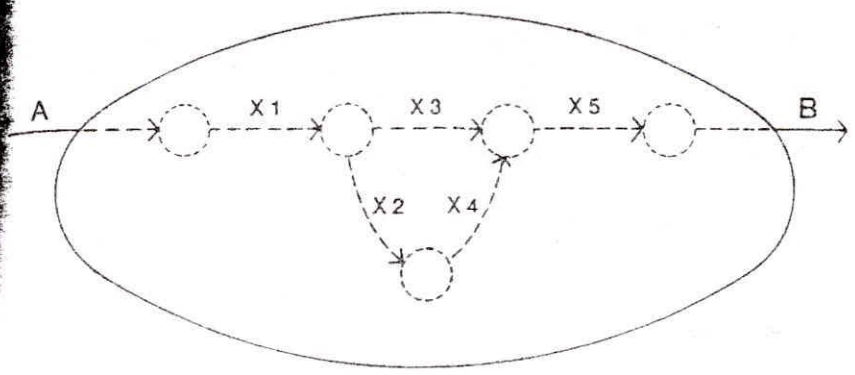


Figura 5.23: Subfunciones y flujos de datos intermedios

En esta tarea de incorporar burbujas, las cuales corresponden a los subprocesos de la función, no es necesario detenerse aún en la asignación de nombres a cada una de estas subfunciones. Lo que sí es importante es la verificación de los flujos de datos internos en cada proceso identificado, puesto que de haberlos y una vez chequeada su existencia con los usuarios, el proceso aparecerá conformado por varios subprocesos, los cuales estarán relacionados por los flujos de datos identificados tal como lo muestra la figura 5.24.

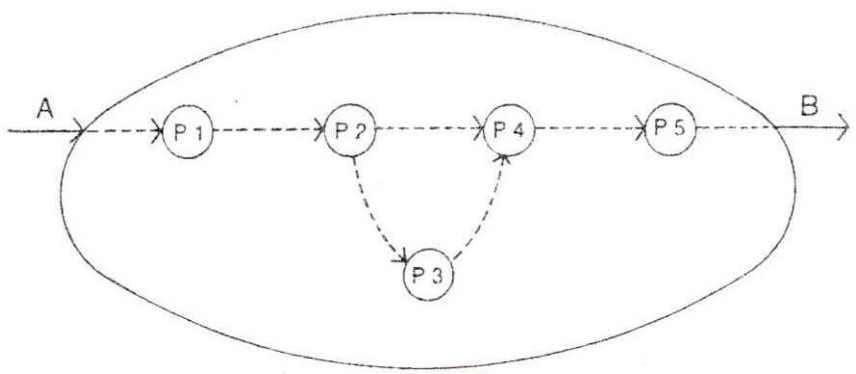


Figura 5.24: Subprocesos relacionados

A continuación, se hace necesario verificar la consistencia de los flujos, esto significa que para cada uno de los flujos generados por algún subproceso es preciso determinar que ítemes elementales de datos se necesitan para conformarlos como tales, y si es posible, con los datos de entrada y las transformaciones a que estos se someten, su obtención.

5.4.4. Asignar nombre a los flujos

Los nombres asignados a los flujos de datos tienen indudablemente un impacto significativo en la calidad y efectividad de la representación, por cuanto la eficacia del modelo dependerá fundamentalmente de la capacidad que posee para reflejar lo más exactamente la realidad que pretende describir. En virtud de ello se sugieren las siguientes recomendaciones:

- Deben asignarse nombres a todos los flujos de datos.
- El nombre debe ser equitativo, esto significa que es preciso aplicarlo a la totalidad del contenido del flujo y no a alguno o algunos de sus componentes.
- Se debe evitar incluir palabras como dato e información, fundamentalmente por su carácter redundante.
- No se debe agrupar ítems dispares o que no pueden tratarse en conjunto como un solo flujo.
- Si hay cierto flujo difícil de nominar, es probable que se trate de un problema de integración inadecuada. En esos casos es conveniente considerar la posibilidad de particionar.

5.4.5. Asignación de nombre a los procesos

Los nombres a los procesos o funciones deben asignarse una vez que se ha asignado nombre a todos los flujos de datos.

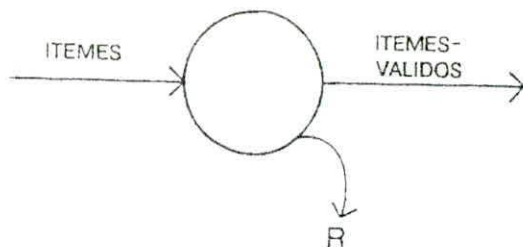


Figura 5.25: Ejemplo de asignación de nombre a un proceso

En la figura 5.25, se puede observar lo natural y simple que resulta asignar nombre al proceso. En este caso, es bastante obvio que el nombre más adecuado es validar-ítems.

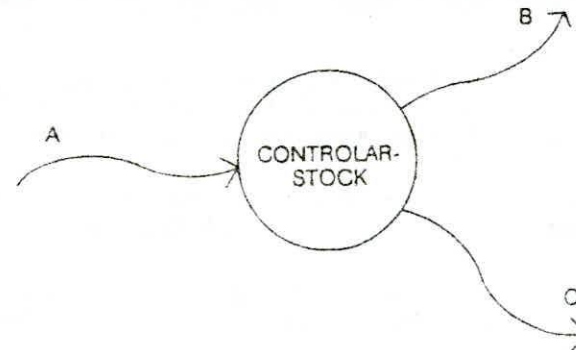


Figura 5.26: Asignación de nombre a los flujos

Por el contrario, si se observa la figura 5.26, se verá que no es fácil asignar nombre a los flujos de datos que llegan y salen del proceso Controlar-stock, si no se tiene la información explícita que indique el contenido de los flujos A, B y C.

Al igual que en el caso de la asignación de nombres a los flujos de datos, para los procesos se pueden sugerir algunas recomendaciones:

- Se debe asignar nombre a todos los procesos.
- Los nombres deben ser equitativos, lo que significa que si un proceso realiza más de una operación ese "más" debe reflejarse en el nombre.

Cabe señalar que al momento de proponer un nuevo sistema el cual se refleja en el diagrama de flujos de datos lógico nuevo, los procesos deben manifestar una alta cohesión,* razón por la cual si el proceso contempla más de una función, ello debe significar que una sin la otra no pueden llevarse a cabo.

Por ello se sugiere que tenga una sola acción verbal y un sujeto en singular. Si hay dos verbos y la cohesión es insuficiente es conveniente pensar en la opción de particionar.

- Se recomienda no usar palabras poco significativas, como proceso o manejo, fundamentalmente por su carácter redundante.
- Si un proceso resulta difícil de nominar, hay que observar su cohesión, y si ella no es adecuada, se sugiere pensar en particionarlo.

*Por cohesión se entiende la medida de asociación funcional de los elementos de una función.

5.4.6. Incorporación de archivos

Una vez conformado el modelo del sistema en función de los flujos de datos que ingresan a él, de los flujos de datos internos, de las transformaciones a que se someten y de los flujos de datos de salida, es preciso incorporar los archivos -entendidos éstos como depósitos temporales de datos- en cada uno de los subprocesos que los utilizan. Para ello hay que conectarlos a cada subfunción que los use mediante flujos que especifican el sentido de los datos que se trasladan del archivo al proceso y viceversa.

5.4.7. Verificación del modelo conformado

El modelo logrado tras estos procedimientos se puede considerar como una primera versión, haciéndose necesario comprobar si se ajusta efectivamente al sistema que se pretende reflejar, a objeto de conformar un diagrama de flujo de datos realmente representativo.

En este sentido, si se ha olvidado incluir un flujo de datos de entrada, éste debe agregarse en la periferia del diagrama y de ser necesario, se debe modificar los límites del sistema.

Del mismo modo, hay que eliminar los flujos de datos que no se utilizan efectivamente en el sistema, ya que si esto ocurre implica que carecen de significado.

Si sucede que el sistema incluye una o más de una malla de procesos completamente desconectada del resto, entonces es preciso estudiar cuidadosamente el sistema representado a objeto de verificar esta desconexión. Si es efectivamente así, es aconsejable explicitar más de un sistema, cada uno de ellos delimitado rigurosamente y establecido en términos de una cohesión significativa.

5.5. Jerarquización de los Diagramas de Flujo de Datos

La figura 5.27, muestra una visión detallada de un sistema, lo cual significa que aparece representado en términos de sus subprocesos o subfunciones de último nivel.

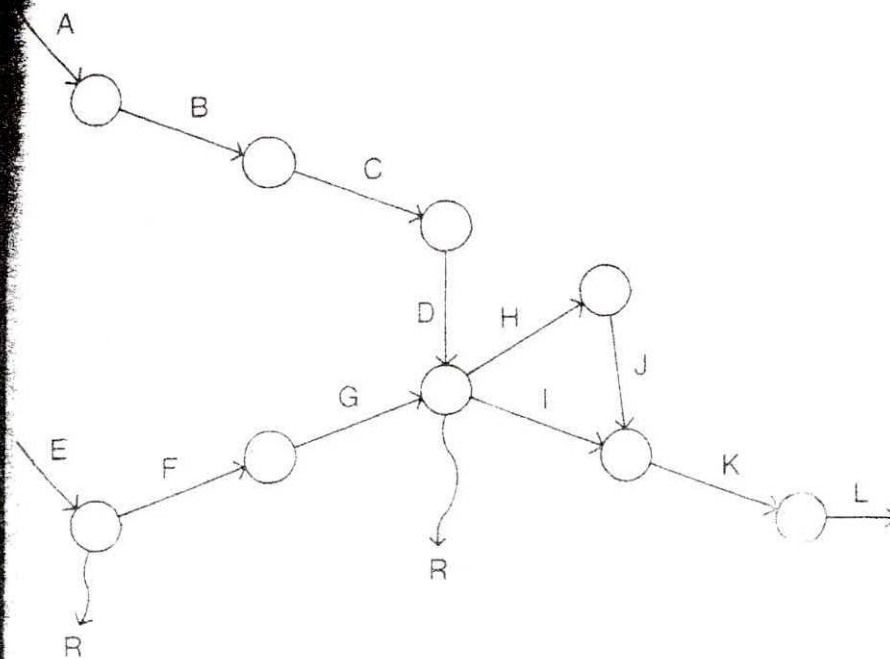


Figura 5.27: Subprocesos de último nivel

Es habitual que a este nivel de la jerarquía, la descripción del sistema involucre una cantidad apreciable de subprocesos y flujos de datos, lo cual hará que el grado de comprensión que se pueda obtener del diagrama se vea considerablemente disminuido.

Para evitar esta dificultad, el diagrama se particiona jerárquicamente, obteniendo lo que se conoce como diagrama jerárquico de flujo de datos. Esto significa que el sistema se representa en diagramas independientes en función de los diferentes niveles que lo describen.

La representación por niveles, en consecuencia, consiste en reflejar el sistema en diagramas que van desde lo más global a lo más detallado, en donde lo más global es la descripción sólo a nivel de entradas y salida, y lo más detallado, es el máximo despliegue de las subfunciones susceptibles de obtener tras el proceso de refinamiento sucesivo de la función global.

El refinamiento sucesivo se traduce en la generación, por cada subfunción identificada, como componente de una función más global, de un conjunto de nuevas subfunciones, las cuales a su vez pueden volver a descomponerse en los mismos términos tal como lo muestra el modelo de la figura 5.28.

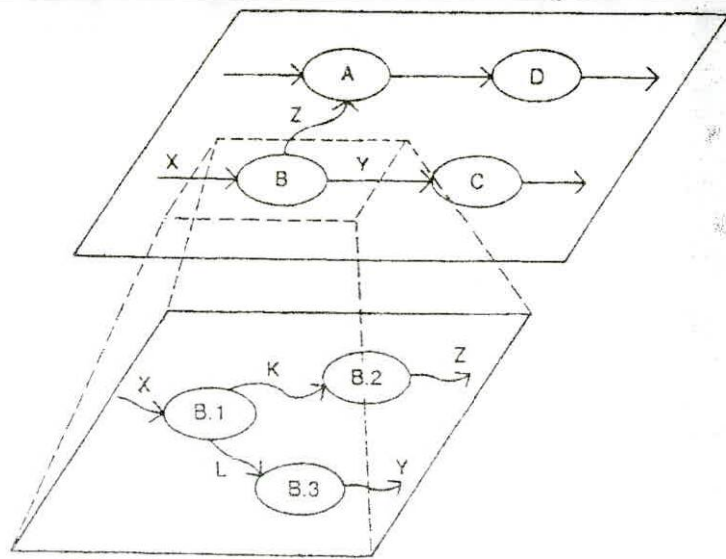


Figura 5.28: Jerarquización de los DFD

En consecuencia, el diagrama jerárquico de flujo de datos representará el sistema en términos de los siguientes diagramas:

- Diagrama de Contexto.
- Diagramas de Funciones Elementales.
- Diagramas de Niveles Intermedios.

3.5.1. Diagrama de Contexto

El diagrama global o de contexto, muestra las entradas y salidas del sistema y las funciones relacionadas, delimitando explícitamente las fronteras del sistema. La figura 5.29 muestra el diagrama de contexto para el diagrama de la figura 5.27.

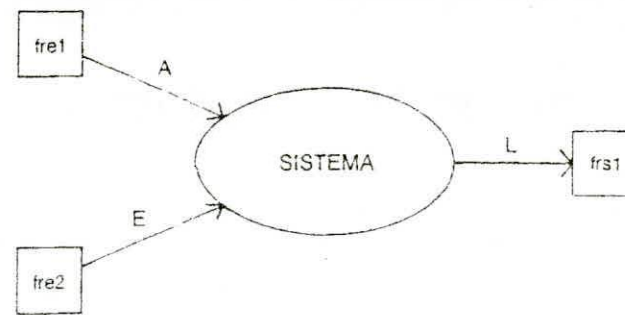


Figura 5.29: Diagrama de contexto

Los diagramas de contexto describen el sistema como una sola burbuja, por lo tanto se trata de una representación no particionada. Generalmente se les somete a una prueba de coherencia, es decir, se verifica la relación lógica que existe entre las entradas y salidas. Para ello, en primer lugar, es preciso considerar el diagrama como una transformación del tipo entrada-transformación-salida, y verificar si a partir de la aplicación de la transformación a la entrada es posible obtener la salida. Es decir, se debe establecer si la transformación es posible.

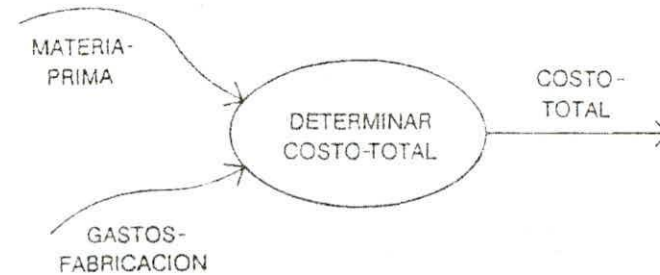


Figura 5.30: Proceso con flujos insuficientes

La figura 5.30 muestra una porción de un diagrama de flujo de datos referido a un sistema de producción a pedido, en donde se necesita determinar el costo total del trabajo realizado. Para ello se requiere de un informe de la materia prima utilizada, de la mano de obra destinada al trabajo y de los gastos de fabricación en que se incurre durante el proceso de fabricación y que son imputables a dicho proceso productivo.

Al observar el diagrama es posible apreciar que sólo con los datos relativos a las materias primas y a los gastos de fabricación no es posible obtener los costos totales, puesto que se carece de la información respecto de la mano de obra. Por lo tanto, en este caso, es necesario revisar los flujos de entrada.

5.5.2. Diagrama de niveles intermedios

Los diagramas de niveles intermedios son todos aquellos que, para ser comprendidos, requieren de una o más particiones.

Corresponden, por lo tanto, a los diagramas obtenidos por refinamiento sucesivo del diagrama de contexto. Estos son, por nivel de descomposición:

- El diagrama de primer nivel de desagregación. Se denomina diagrama cero y es un diagrama de segundo nivel. Es decir, corresponde a una primera desagregación de la función o proceso y representa el sistema en términos de las subfunciones globales que conforman su estructura funcional. La figura 5.31 muestra el diagrama cero para el sistema de la figura 5.27. En estos diagramas las burbujas generadas, además de llevar un nombre deben numerarse correlativamente.

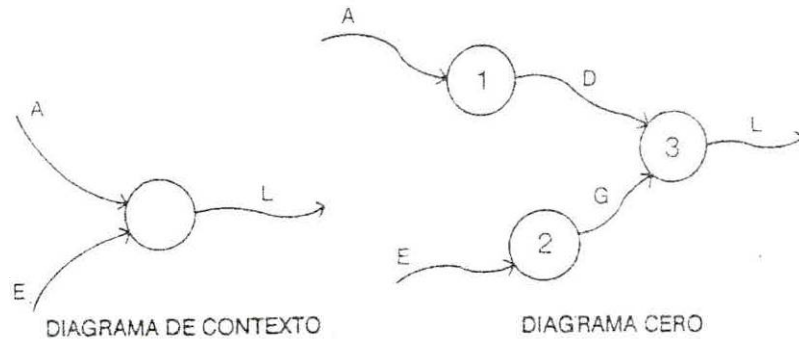


Figura 5.31: Diagramas de niveles superiores

- Los diagramas de segundo nivel de desagregación. Corresponden al refinamiento de cada uno de los subprocesos identificados en el diagrama cero. Se obtendrá, en consecuencia, tantos diagramas como burbujas haya en el primer nivel, tres para el sistema de la figura 5.31, tal como aparece en la figura 5.32.
- Los diagramas de tercer nivel de desagregación y sucesivos. Corresponden al particionamiento de cada una de las subfunciones del nivel inmediatamente superior. En consecuencia, también en ellos habrá tantos diagramas derivados como funciones haya en los diagramas del nivel superior del cual se obtienen.

En todos los diagramas de los niveles intermedios cada uno de ellos se identifica con el número del subproceso que describe mediante particionamiento. Por ejemplo, un diagrama 3.1 describe los subprocesos y relaciones del proceso 3.1 del diagrama 3.

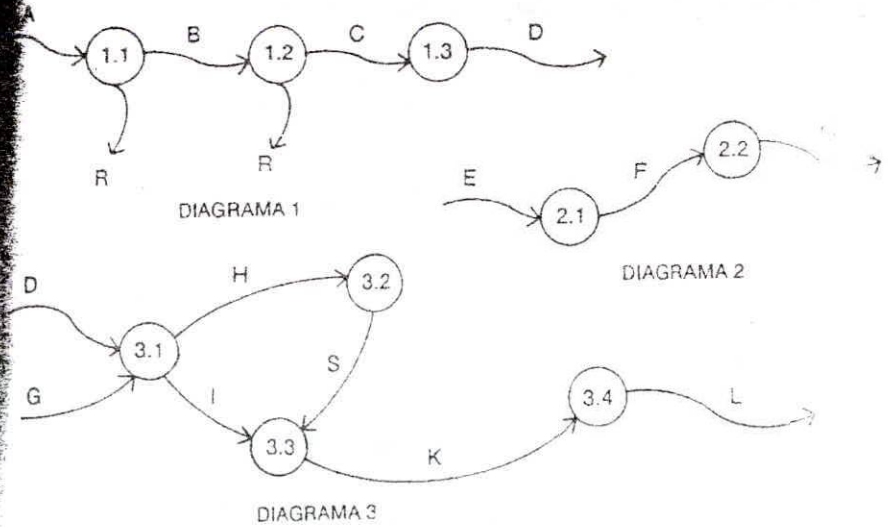


Figura 5.32: Descomposición de un diagrama cero

En resumen, los diagramas jerárquicos de flujo de datos se conforman en todo a un modelo que muestra el sistema refinado sucesivamente, lo cual significa que cada proceso en el diagrama (las burbujas) debe ser desagregado posteriormente para dar más detalles acerca de los subprocesos que lo conforman. La figura 5.33 ilustra los alcances de este particionamiento.

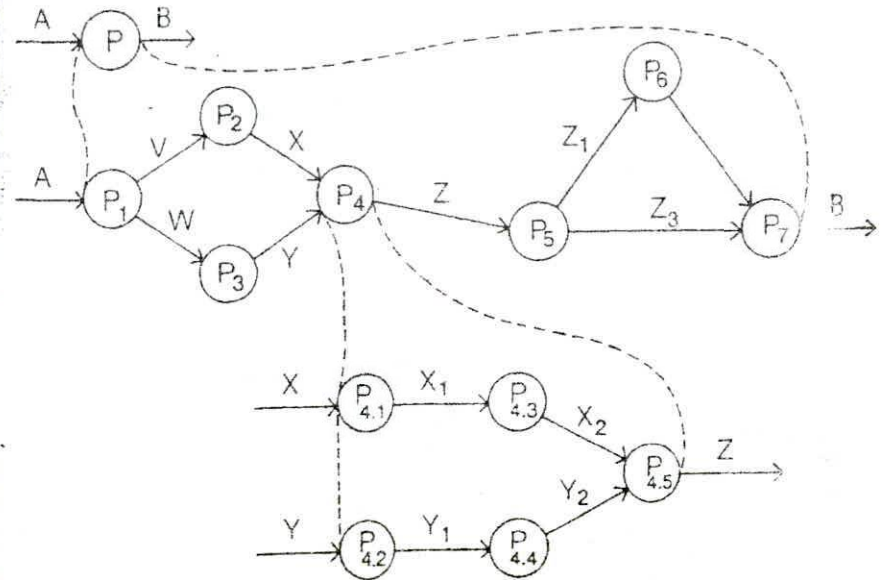


Figura 5.33: Particionamiento de los procesos

Generalmente las relaciones entre las burbujas de un nivel inferior y las de un nivel superior se identifican como relaciones padre-hijo. En la figura 5.34, el diagrama dos es un diagrama hijo del diagrama cero.

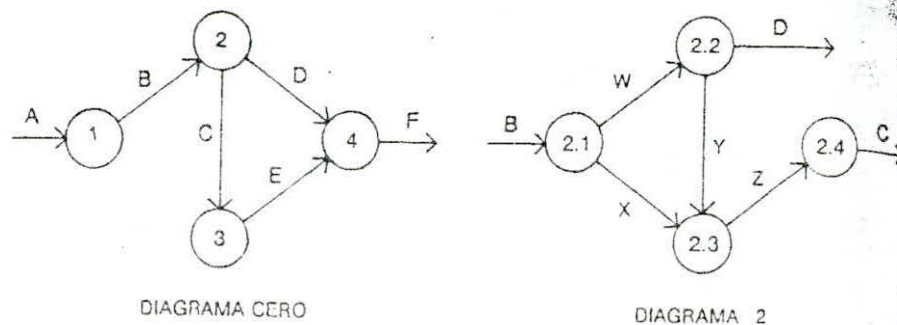


Figura 5.34: Diagramas padre-hijo

5.6. Convenciones de desagregación

La visión más global del sistema se representa mediante el diagrama de contexto.

Este primer nivel estructural a su vez se desagrega dando origen a un diagrama de segundo nivel, el diagrama cero, que corresponde a una primera desagregación de la función o proceso, y que está compuesto por el conjunto de subprocesos derivados de este primer refinamiento. En consecuencia, si el diagrama cero, en cuanto diagrama de primer nivel de desagregación es DN1, sus componentes serán:

$$DN1 = [sp1, sp2, \dots, spi, \dots, spn]$$

donde spi : i-ésimo subproceso, y

spn : n-ésimo subproceso

n : número de subprocesos del diagrama cero.

Este diagrama cero, a su vez, se refina dando origen a los diagramas de segundo nivel -DN2- los cuales serán tantos como subprocesos haya en el diagrama cero. Es decir:

$$DN2 = [D1, D2, D3, \dots, Di, \dots, Dn]$$

donde Di : i-ésima burbuja del diagrama cero, y

n : número de diagramas del segundo nivel.

A su vez, cada uno de los diagramas de segundo nivel se desagregan dando origen a los diagramas de tercer nivel. Estos serán tantos como burbujas haya en los diagramas de segundo nivel, es decir, habrá:

$$DN3 = \begin{bmatrix} D1.1 & D1.2 & \dots & D1.j & \dots & D1.m_1 \\ D2.1 & D2.2 & \dots & D2.j & \dots & D2.m_2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ Di.1 & Di.2 & \dots & Di.j & \dots & Di.m_i \\ \dots & \dots & \dots & \dots & \dots & \dots \\ Dn.1 & Dn.2 & \dots & Dn.j & \dots & Dn.m_n \end{bmatrix}$$

donde

$D1.1$: 1ª burbuja del diagrama 1,

$Di.j$: j-ésima burbuja del diagrama i,

m_1 : número de burbujas del diagrama 1,

m_i : número de burbujas del diagrama i,

$Di.m_i$: m_i -ésima burbuja del diagrama i, y

m_n : número de burbujas del diagrama n.

Por ejemplo, si se tiene el diagrama cero de la figura 5.35-a, se debe generar, a partir de él, como diagramas de segundo nivel, cuatro diagramas tal como el diagrama tres de la misma figura. Esto significa que el subproceso 3 se puede explicar, estructuralmente en términos los cuatro subprocesos de la figura 5.35-b.

DIAGRAMA CERO

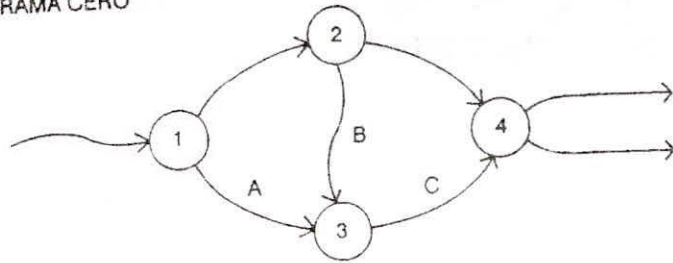


Figura 5.35 a

DIAGRAMA TRES

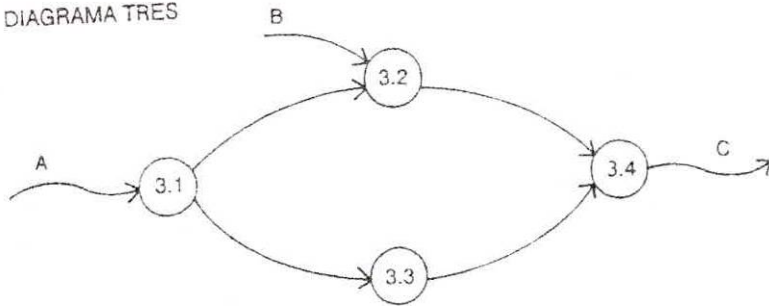


Figura 5.35 b
Desagregación de los procesos

5.6.1. Balanceo de mallas

Si se observa la figura 5.35-a, es posible que, en el diagrama cero, el proceso 3 tiene dos entradas, los flujos A y B y una salida, el flujo C. Esta entrada y las dos salidas aparecen también en el diagrama tres de la figura 5.35-b.

Los flujos de datos que entran y salen de una burbuja en un diagrama padre, deben ser equivalentes a las Entradas y salidas en un diagrama hijo.

Si se tienen los diagramas de flujo de datos de la figura 5.36-a, se puede apreciar que ellos no están balanceados, ya como entrada en el diagrama hijo falta el flujo M, y como salida, sobra el flujo S.

La única excepción a la regla del balanceo se presenta cuando el proceso contempla rechazos.

Por ejemplo, si en un proceso de validación del contenido de un flujo de datos, este se rechaza por no ser válido, este flujo no se considera para evaluar el balanceo de las mallas, como lo muestra la figura 5.36.

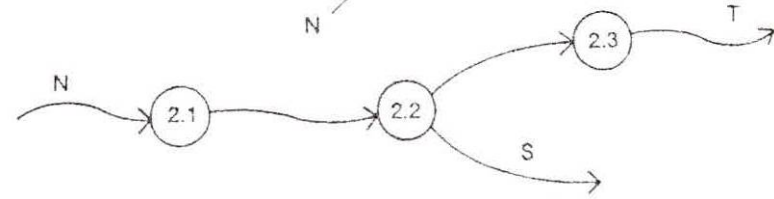
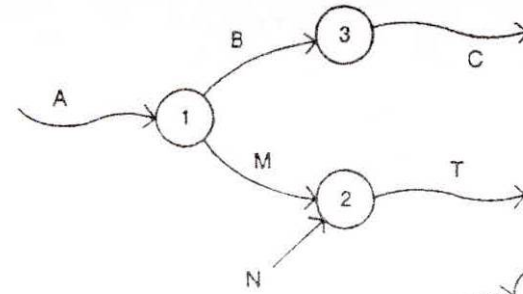


Figura 5.36 a

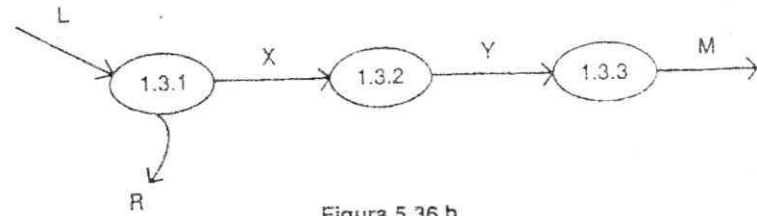
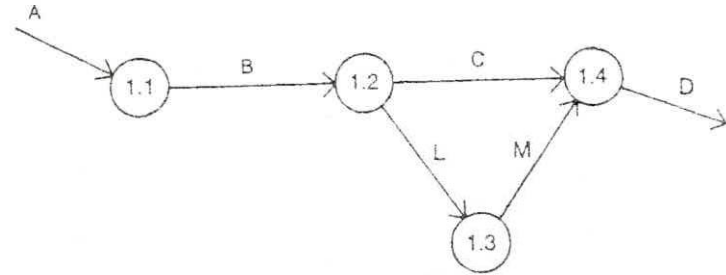


Figura 5.36 b
Balanceo de DFD

A veces, el verificar el balanceo requiere de un análisis que va más allá de la simple observación de los diagramas padre e hijo.

Por ejemplo, si se tiene los diagramas de la figura 5.37. En ellos se aprecia que no están los mismos flujos en el diagrama padre (diagrama cero) y en el diagrama hijo (diagrama 3). Por lo tanto, la conclusión obvia al observar los diagramas es que se trata de un desbalanceo, sin embargo ello no es así. Por

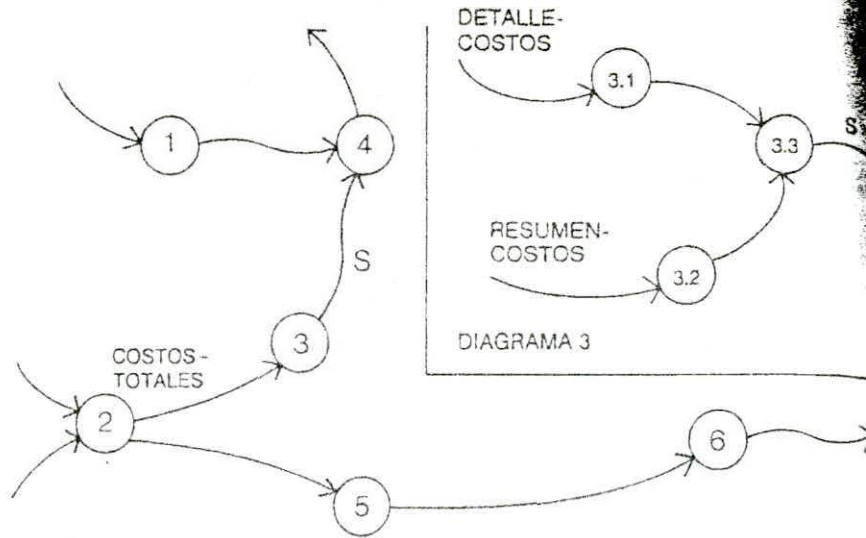


DIAGRAMA CERO

Figura 5.37: Desbalanceo aparente entre los DFD

eso es conveniente ir más allá en el análisis, ya que podría existir lo que se denomina equivalencia neta, lo cual ocurre, en este caso, si:

$$\text{COSTOS TOTALES} = \text{DETALLE COSTOS} + \text{RESUMEN DE COSTOS}$$

Para verificar, en estos casos el cumplimiento de las reglas del balanceo, es preciso recurrir al diccionario de datos.

Lo que ha ocurrido es que en el diagrama hijo hay una representación más detallada que en el diagrama padre, tanto a nivel de funciones como de flujos de datos, es decir, existe una descomposición paralela de datos y proceso.

Si en el diccionario de datos, se explicita que el flujo del diagrama padre está compuesto exactamente por los flujos del diagrama hijo, se puede afirmar que existe balanceo.

6.2. Convenciones respecto de la numeración

Para cualquier burbuja el número que la identifica se conforma de acuerdo al siguiente orden:



En donde "número del diagrama", es el número del diagrama padre, y "número local" es un número correlativo que refleja la secuencia o el orden en que el subproceso lleva a cabo su tarea, dentro del proceso que lo comprende. Cabe hacer notar, que el número es solamente referencial, y puede asumir cualquier orden arbitrario. Si se considera el hecho de que los diagramas de flujo de datos son un modelo, en estado estacionario, del sistema que se está modelando, entonces no tiene sentido intentar descubrir un orden en los subprocesos. La figura 5.38 muestra un diagrama cero con tres subprocesos:

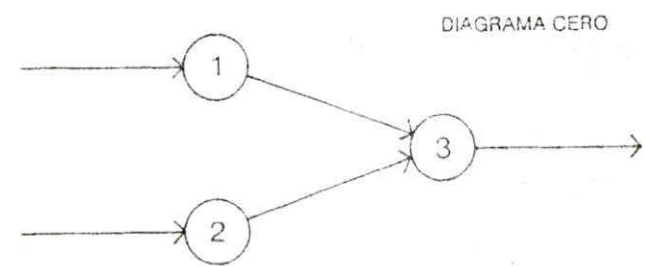


Figura 5.38: Numeración correlativa en un diagrama cero

Rigurosamente, en este caso la numeración debería ser como lo muestra la figura 5.39, sin embargo por convención se numeran simplemente 1, 2 y 3.

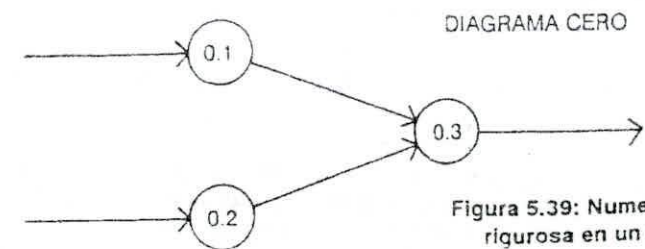


Figura 5.39: Numeración correlativa rigurosa en un diagrama cero

En consecuencia para un diagrama uno la numeración de sus burbujas se como lo muestra la figura 5.40.

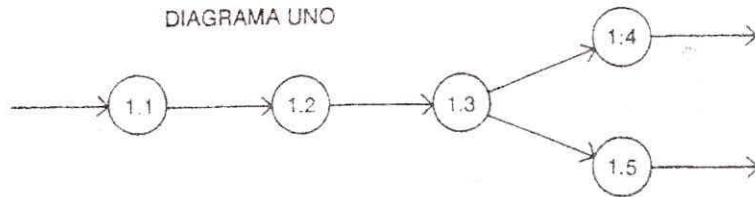


Figura 5.40: Numeración correlativa en un diagrama uno

Existe también la posibilidad de una numeración abreviada. En este caso, si se tiene, por ejemplo un diagrama 5.8.4, sus burbujas deben numerarse (5.8.4.1); (5.8.4.2) (5.8.4.n). Sin embargo, para evitar lo engorroso de anotar al interior de ellas números tan extensos se usa la notación abreviada en los términos que muestra la figura 5.41.

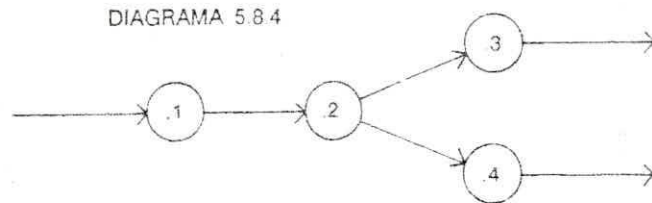


Figura 5.41: Numeración abreviada

5.6.3. Ubicación de los archivos en diagramas jerárquicos

Los archivos deben presentarse en el nivel en que son realmente utilizados por los diferentes procesos, es decir, en donde efectivamente se realiza sobre ellos operaciones de ingreso y extracción de datos.

Si se tiene el diagrama 4 y su diagrama hijo 4.3, tal como lo muestra la figura 5.42, se puede apreciar que el archivo ALFA es utilizado sólo por los procesos 4.3.1 y 4.3.2. No podría por lo tanto aparecer en el diagrama 4 ya que no todos los subprocesos del proceso 4 hacen uso del archivo ALFA. En consecuencia, debe aparecer en el diagrama 4.3, que es aquel en el cual se describen los subprocesos que efectivamente hacen uso del archivo, en este caso particular, los subprocesos 4.3.1 y 4.3.2.

Es la misma razón por la cual en el diagrama 4 no aparece el flujo XXX puesto que éste es atingente sólo a los subprocesos 4.3.1 y 4.3.3 del diagrama 4.3

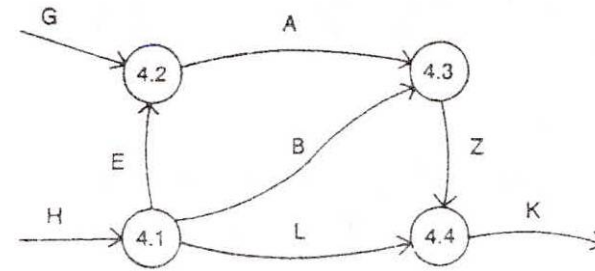


DIAGRAMA 4

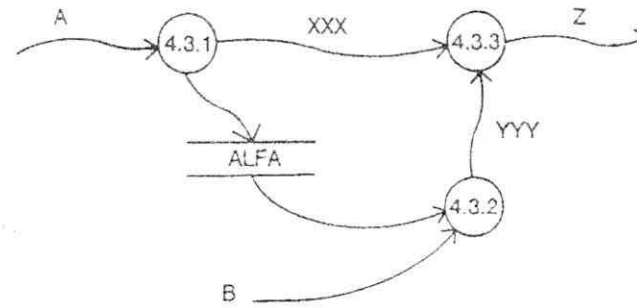


DIAGRAMA 4.3

Figura 5.42: Representación de archivos en los DFD

5.6.4. Ventajas de los Diagramas de Flujo de Datos Jerárquicos

Entre las ventajas de los diagramas jerárquicos de flujo de datos se puede citar:

- La representación modular del sistema hace que la incorporación de cambios posteriores, es decir, la mantención del sistema, a través de los diagramas de flujo de datos, sea más fácil de llevar a cabo.
- La presentación del sistema en un esquema "top-down" facilita su lectura, con lo cual los diagramas de flujo de datos se constituyen en un instrumento de mucha versatilidad. Por ejemplo, si un ejecutivo desea tener una visión global de un sistema podrá remitirse a los niveles superiores del diagrama. Un empleado, seguramente más interesado en aquellos procedimientos que se relacionen con su labor, recurrirá preferentemente a los niveles inferiores y a sus especificaciones.

- Los diagramas jerárquicos de flujo de datos en cuanto modelo del sistema facilitan la interacción usuario-analista, al constituirse, por su carácter gráfico, en un instrumento eficaz de comunicación.

5.6.5. Origen y destino de los flujos de datos en los diagramas intermedios

A objeto de no disminuir la eficacia de los diagramas de flujo de datos, en cuanto instrumento de comunicación, se aconseja no incluir en los diagramas hijo los orígenes y destinos ya que ellos complican el diagrama, al incluir elementos que en nada ayudan a la descripción del sistema.

Si se desea saber, por ejemplo, el destino de un flujo de salida basta con remitirse al diagrama padre para obtener dicha información.

Por ejemplo en la figura 5.43, en el diagrama 3 no se tiene información respecto del destino de los flujos d y e, sin embargo, en el diagrama padre, en este caso el diagrama cero, se puede apreciar que el destino del flujo d es el proceso 4 y el destino del flujo e, el proceso 5.

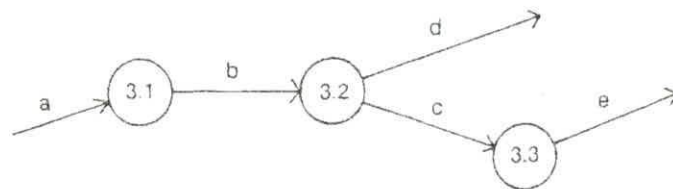


DIAGRAMA 3

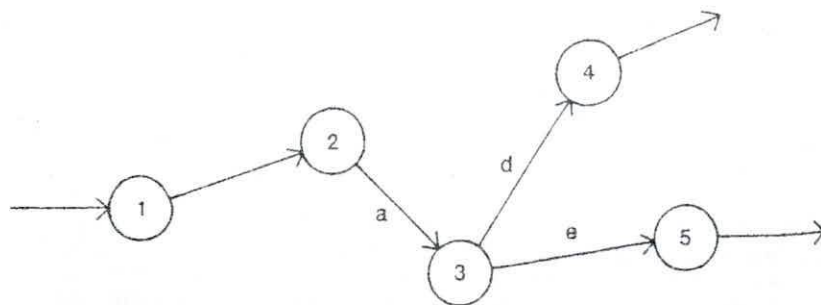


DIAGRAMA CERO

Figura 5.43: Origen y destino de los flujos de datos intermedios

5.6.6. Extensión de la desagregación

Los diagramas deben explicarse a sí mismos. Si en ellos hay muchas burbujas, la gráfica se torna incomprensible.

Por evidencias empíricas se ha determinado como óptimo incluir, en cada diagrama, 7 ± 2 burbujas [MIL-56].

5.6.7. Consideraciones respecto del último nivel de desagregación

El último nivel de desagregación establece el término del proceso de refinamiento sucesivo.

El proceso de refinamiento sucesivo se detiene cuando no es posible identificar, en el nivel inmediatamente inferior, flujos de datos explícitos.

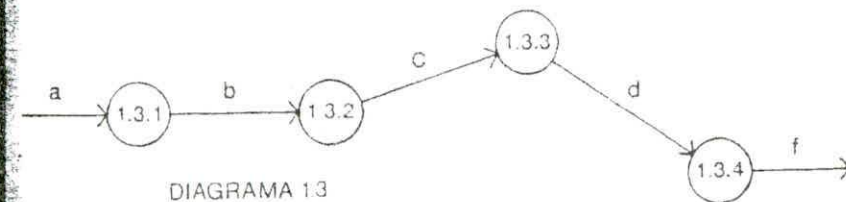


DIAGRAMA 1.3

Figura 5.44: Último nivel de desagregación en los DFD

En otras palabras, se detiene cuando al interior de las burbujas derivadas no se encuentran flujos de datos documentados que deban ser transformados para generar otro u otros flujos de datos. Es decir, cuando ninguno de los flujos a, b, c o d de la figura 5.44 es transformado al interior de la burbuja, a la cual ingresan, en flujos intermedios antes de dar paso al flujo de salida.

En estos casos, la transformación del flujo de datos de entrada en flujo de datos de salida se lleva a cabo en virtud de ciertos procedimientos específicos que rigen la transformación.

En consecuencia, para las burbujas de último nivel, identificadas normalmente como primitivas o elementales no existe un diagrama de flujo de datos adicional, sino que una especificación del procedimiento que rige la transformación y se documenta mediante el uso del Español Estructurado -un subconjunto del idioma español-, Tablas de Decisión o Árboles de Decisión.

5.6.8. Evaluación y refinamiento de los diagramas de flujo de datos

No es fácil obtener un diagrama de flujo de datos rigurosamente correcto en la primera versión. Ello sugiere la necesidad de someterlo a diferentes pruebas para refinarlos y obtener, a la postre, versiones más exactas en relación a la realidad existente o esperada, que se pretende reflejar con los diagramas de flujo de datos.

Las técnicas sugeridas para ello son diversas, no obstante cabe destacar de ellas, que son fundamentales: las pruebas de exactitud y la pruebas de efectividad.

Las pruebas de exactitud son aquellas que se aplican para verificar la correspondencia del diagrama de flujo de datos con la realidad que se desea representar; más específicamente, con el sistema, que los responsables de la función o proceso organizacional sometida a tratamiento informático, desean que se construya.

Las pruebas de efectividad son aquellas que se aplican para verificar la calidad del diagrama de flujo de datos como instrumento de comunicación.

5.6.8.1. Pruebas de exactitud

Por lo general, los primeros diseños de un diagrama de flujo de datos que describen un cierto sistema, no se ajustan exactamente a la realidad. Generalmente son incorrectos, no funcionan, no toman en cuenta las características reales y no calzan. La razón por la cual se configura un diagrama de flujo de datos, sabiendo que es erróneo, radica en que la obtención de un modelo exacto es un proceso paulatino, lo cual significa que las primeras versiones no ajustadas a lo que realmente se espera, se van perfeccionando gradualmente hasta alcanzar diagramas más exactos.

El proceso mediante el cual se van configurando diagramas cada vez más exactos, se centra en la observación de algunos aspectos fundamentales que no deben dejar de revisarse. Estos son:

- Revisión de los flujos de datos: es posible que falten flujos de datos, cuyo contenido se requiera en algún proceso. O bien puede ocurrir que existan flujos de datos innecesarios, cuyo contenido no se requiere en proceso alguno.
- Revisión de los procesos: es posible que falten procesos o que algunos estén incorrectamente jerarquizados, como también que los mismos estén inadecuadamente denominados.

Otras revisiones: puede que existan flujos de control que es preciso eliminar. Incluso puede ocurrir que el diagrama de flujo de datos no corresponda exactamente a la realidad. Aunque no es fácil detectar este tipo de error, es conveniente no dejar de considerar esta posibilidad.

Errores relativos a la documentación de los diagramas de flujo de datos

Una deficiencia común de los diagramas de flujo de datos es la existencia de flujos de datos sin nombre. Si ello ocurre significa que no existe claridad total acerca del contenido de algunos flujos del diagrama de flujo de datos. Cuando se intenta poner nombre a algún flujo y no se logra hacerlo, ello se debe, probablemente, al desconocimiento de los datos que contiene. Los flujos de datos sin nombre generalmente son consecuencia de la aplicación incorrecta de la metodología de análisis estructurado; más específicamente, del error de centrar la atención primero en las funciones y luego en los flujos de datos. La figura 5.45 pone de manifiesto estas deficiencias. En dicho diagrama de flujo de datos no se ha especificado una interfase; seguramente porque se incorporó al diagrama ciertas funciones sin saber si entre ellas existe un flujo de datos explícito, o se ignora cuál es el flujo que las relaciona. Muchas veces esto significa que se ha particionado indebidamente una subfunción, la cual, por su cohesión interna, hace que no existan flujos de datos explícitos con alcances interfases.

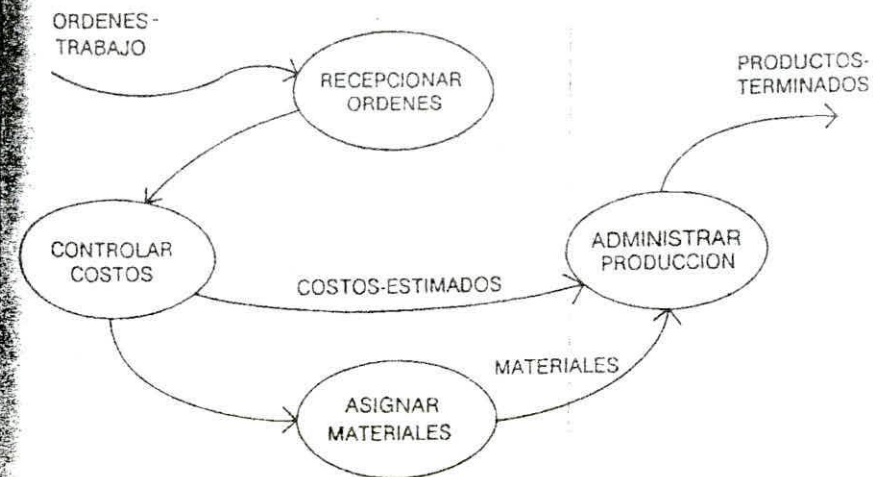


Figura 5.45: Errores en la documentación de los flujos de datos

También la figura 5.46 muestra un flujo de datos sin nombre. En este caso se trata específicamente de la utilización de pseudonombres, como por ejemplo información requerida. Este es un flujo sin significado real, seguramente para ocultar la ignorancia respecto de los datos que fluyen por él. Obviamente nombres como datos, información, entrada y salida no tienen significado en un diagrama de flujo de datos. Una forma de asegurar el conocimiento efectivo que se tiene respecto de un flujo de datos es definirlo como ítem del diccionario de datos puesto que para hacerlo es preciso conocer su composición y estructura.

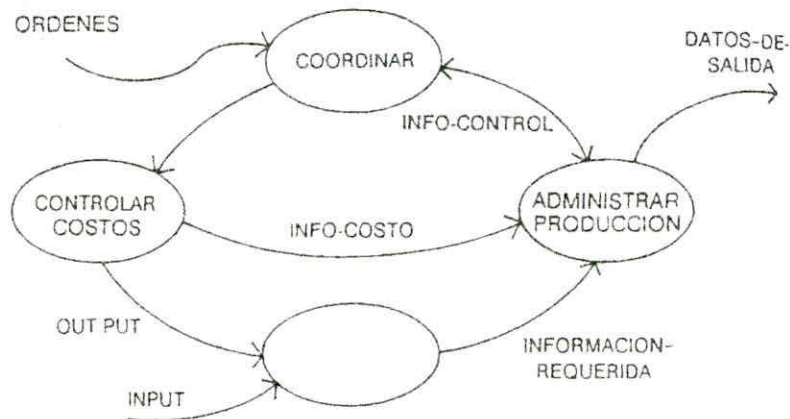


Figura 5.46: Errores en los nombres de los flujos

También esta tarea de definir los flujos de datos en el diccionario de datos permite eliminar los pseudoflujos de datos, como por ejemplo: señal-de-inicio que aparece en la figura 5.47, señal-de-inicio es un indicador de control que especifica el momento en el cual se debe iniciar una cierta operación en un determinado proceso. Se sabrá que se trata de una instancia de control tan pronto como se intente definir su composición, simplemente porque carecen de ella. Estos pseudoflujos de datos deben eliminarse como flujos de datos del diagrama de flujo de datos, puesto que corresponden más bien a las especificaciones de los procedimientos en virtud de los cuales los procesos deben llevar a cabo su tarea.



Figura 5.47: Seudoflujos

También es necesario prestar atención a los nombres asignados a las burbujas. En relación a ellas, primero hay que verificar si existe correlación entre el nombre y los flujos de datos de entrada y salida. En la figura 5.48 por ejemplo el nombre del proceso no guarda relación con el flujo "alumnos-reprobados".

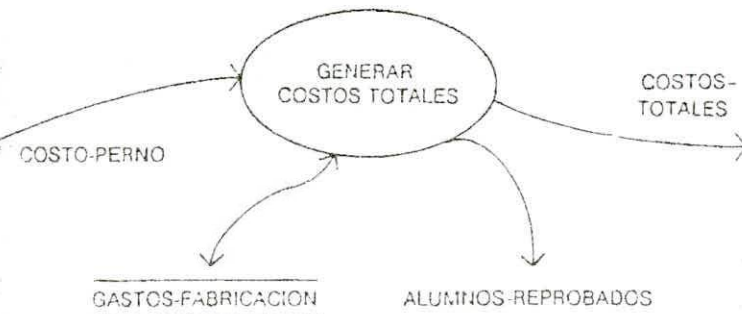


Figura 5.48: Nombre de proceso mal asignado o flujo incorrecto

En segundo lugar, hay que verificar que el nombre refleje con exactitud todo lo que el proceso lleva a cabo, incluyendo aquello que realizan sus nodos hijos. Se trata de evitar que los nombres causen confusión, al ocultar o distorsionar lo que realmente realiza la función localizada en la burbuja.

En resumen, es preciso verificar que todos los flujos de datos tengan nombre, verificar que todos los nombres de los flujos sean definibles, eliminar todos los flujos de datos que tengan composición nula, verificar la consistencia de los nombres de las burbujas con las respectivas entradas y salidas, y verificar que los nombres de las burbujas involucren sus burbujas hijo.

Errores de consistencia

Una falla común es que los diagramas de flujo de datos estén desbalanceados. Generalmente estos problemas surgen cuando se trabaja en los niveles inferiores. Cuando se agrega o elimina un flujo de datos en un cierto nivel "n", es necesario volver al nodo padre y observar de donde proviene y a donde va. Si ello no se hace, se puede omitir algún cambio que sea preciso incorporar en otra subfunción. Es, en consecuencia, fundamental verificar que se cumpla rigurosamente la regla de balanceo de mallas.

Errores de redundancia

Al modelar el sistema actual, se debe tener especial cuidado con las subfunciones componentes de cada función más global, o que estén en la frontera linda con otra función.

Es habitual que los responsables de cada una de dichas subfunciones globales las consideren como parte de su área de interés, por lo cual podrían aparecer repetidas en dos lugares diferentes. Esta situación se refleja casi siempre como un problema de balanceo, y debe resolverse poniendo la subfunción ya sea en un lugar u otro. También puede suceder lo contrario, es decir, que una subfunción limítrofe no se incluya en ninguna de las funciones globales. Generalmente esto se manifiesta por la existencia de un flujo de datos de salida de una subfunción que no aparece como entrada en la subfunción inmediatamente relacionada.

Carencia de datos requeridos

Si se observa el segmento del diagrama de flujo de datos de la figura 5.49-a, se puede constatar una deficiencia evidente, ya que no todos los datos requeridos para obtener un listado de notas-finales aparece como entrada al proceso. No están las notas parciales. Para obtener las notas finales de una asignatura lo normal es que además de las notas del examen final, se requieren las notas obtenidas durante el período lectivo, tal como lo muestra la figura 5.49-b.

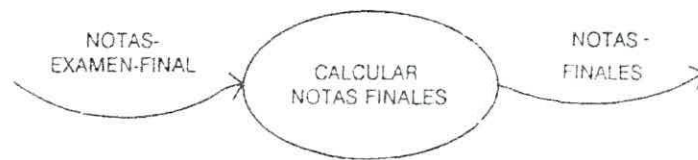


Figura 5.49 a

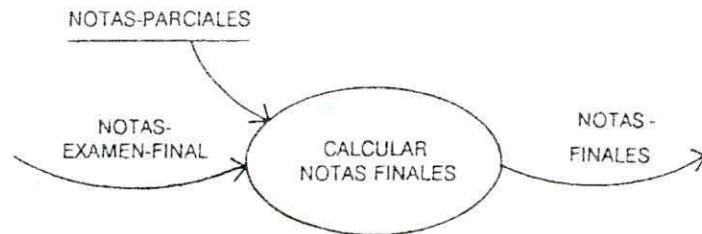


Figura 5.49 b: Problemas con la falta de datos requeridos

La regla de la existencia de los datos requeridos señala que un proceso debe ser capaz de obtener sus salidas usando sólo los datos de los flujos que llegan a él, además ciertos datos constantes. Los procesos que violan esta regla están completamente incorrectos, y en ellos lo más probable es que un flujo de entrada haya obviado. Si esto ocurre, hay que agregar el flujo de datos requerido para mantener el diagrama correcto.

También puede suceder lo contrario, es decir que un cierto flujo de datos simplemente no se utilice en el proceso. Este, aunque no es un error gravísimo, debe evitarse, a objeto de mantener la rigurosidad que, como instrumento de representación, debe manifestar el diagrama de flujo de datos.

Problemas con los archivos

La figura 5.50 muestra una parte de un diagrama de flujo de datos en dos niveles sucesivos. Si se observa con atención el diagrama 4, se puede apreciar en él un error significativo, el archivo DELTA sólo recibe datos, es decir los datos van a él, pero no salen más. Obviamente no tiene sentido mantener almacenados en un archivo datos que jamás se utilizarán.

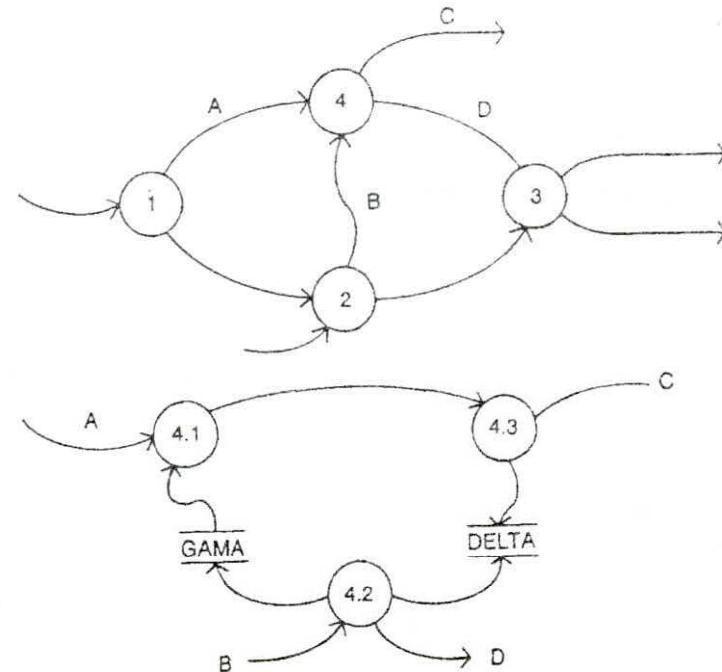


Figura 5.50: Ejemplo de error con los archivos

Si en un diagrama de flujo de datos corresponde al primer nivel en donde se declara un archivo, entonces deben aparecer todas las entradas y salidas de este archivo, es decir un archivo aparece por primera vez en el diagrama en el que se usa. DELTA es local respecto del diagrama 4, por lo tanto todas las referencias a él deben aparecer justamente allí. El diagrama 4 muestra que el archivo no es utilizado. Esto, por cierto, debe ser revisado y corregido. Seguramente verificando el uso del archivo se descubrirá alguna función que no había sido considerada.

Un error común en relación a los archivos es que los flujos que los relacionan a los procesos, no reflejan flujos netos de datos.

Por ejemplo en la figura 5.51 el proceso YYY sólo actualiza el archivo XXX, por lo tanto no debe dibujarse el flujo con doble flecha, ya que ésta se usa en aquellos casos en que, por ejemplo, además de actualizar el archivo, de él se requieren datos que el proceso utilizará para otros propósitos.

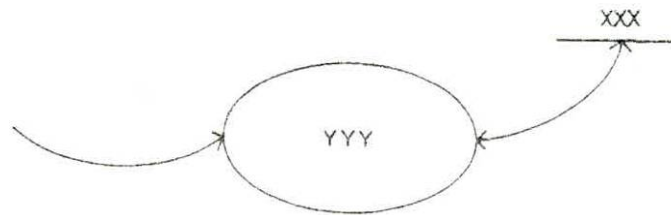


Figura 5.51: Errores en relación a archivos

Errores de concepción

Los errores de concepción se refieren a la configuración de un modelo que no refleja exactamente el sistema que se debe representar.

La única forma de eliminar los errores de concepción es conformar los diagramas de flujo de datos en contacto directo con los involucrados en la función o proceso objeto de análisis, contrastando permanentemente lo realizado para asegurar la exactitud de su representación. La principal herramienta disponible para ello son los "walkthrough" para el análisis (*).

Los "walkthrough" usados durante la fase de análisis son bastante diferentes de aquellos que se utilizan en la fase de implementación. Sin embargo, en ambos casos los aspectos considerados son de importancia significativa para el éxito de cualquier proyecto informático. En la fase de análisis hay una diferencia significativa establecida por la presencia del usuario, por lo tanto los aspectos a considerar se sitúan de una u otra forma en el límite entre dos áreas de responsabilidad: la del usuario y la del analista.

* Walkthrough es la revisión de un producto presentado por su autor realizada por un pequeño grupo de personas para descubrir eventuales fallas

En relación con la aplicación de "walkthroughs" en la fase de análisis Tom De Marco [DEM-79] sugiere las siguientes consideraciones:

Los primeros walkthroughs deben aplicarse en el propio terreno de los usuarios, deben ser muy breves en tiempo y ellos no deben estar expresamente enterados de que participan en ellos.

- No entregar, como avance, los diagramas de flujo de datos a los usuarios, a menos que estén familiarizados con la metodología. Esta nueva metodología debe entregarse a los usuarios bajo estricto control.

- Mostrar a los usuarios nuevos los diagramas de medio y bajo nivel primero. Retener los conceptos de niveles hasta que estén familiarizados con las ideas básicas.

- Complementar el walkthrough de los diagramas de flujo de datos con información física adicional (gente, lugares, nombre de documentos, etc.) para ayudar a que el usuario siga solo.

- A medida que los diagramas de flujo de datos sean presentables, y los usuarios estén más familiarizados con ellos, utilizar walkthroughs formales.

- Mientras se recorra un nivel "n" de un diagrama de flujo de datos, los miembros del equipo de walkthrough deben tener el nivel n-1 en sus manos. Esto les ayudará a mantenerse con seguridad en el contexto de la discusión.

- En todos los intercambios con el usuario, es conveniente evitar el uso de términos especializados para ellos. Un diagrama de flujo de datos debe ser una gráfica o esquema, el diccionario de datos una lista de descripciones de interfases y documentos y la especificación de procedimientos un texto para describir los procedimientos.

5.6.8.2. Pruebas de efectividad

A veces se logra conformar diagramas de flujo de datos técnicamente correctos, pero de poca efectividad, debido a que son de difícil lectura y comprensión.

Para verificar la efectividad de un diagrama de flujo de datos es preciso centrar la atención en:

- La complejidad de las interfases: es decir, revisar las burbujas más complejas (aquellas con muchas entradas y salidas) con el propósito de ver si se pueden separar en dos o más partes o trasladar algunas tareas a otras burbujas.

- Los nombres de los procesos: el nombre ideal para los procesos de los niveles más bajos de desagregación es aquel conformado por una sola forma verbal con un objeto; ello es válido sólo para los niveles de desagregación más bajos. Si un proceso de último nivel tiene más de dos verbos, tal vez haya que particionar o redefinirlo.
- El particionamiento: el particionamiento ideal es aquel que subdivide en partes del mismo tamaño, evitando que en un mismo nivel una burbuja sea elemental y otras se descompongan en más subfunciones derivadas.

Las Tareas del Análisis Estructurado

Referencias

- ACE-88, Acevedo R., G. Cortés, Redes de Petri; una herramienta para el modelado de sistemas de información; Gestión tecnológica, Universidad Técnica Federico Santa María, 1988.
- DEM-79, De Marco, T., Structured Analysis and Systems Specification, Yourdon Press, New York, 1978.
- MIL-56, Miller, G. A., The magical Number Seven, plus minus two: some limits on our capacities for processing information, Psychological Review, Vol. 63, Marzo 1956.

Otras Lecturas

- Freeman, P., Requirements Analysis and Specification, Proc Intl. Computer Technology Conf. ASME, San Francisco, August, 1980.
- Gane, T., Sarson C., Análisis Estructurado de Sistemas, El Ateneo, Buenos Aires, 1977.
- Gehani, N., D. Mc Gettric, eds., Software Specifications Techniques, Addison-Wesley, 1986.
- Martin, J., C. Mc Clure, Diagramming Techniques for Analysts and Programmers, Prentice-Hall, 1985.

En un ambiente de desarrollo de sistemas informáticos, sean estos sistemas de información o de procesamiento de datos y en el contexto de una visión estructurada -como alternativa al enfrentamiento clásico- se distinguen seis actividades principales tal como se puede apreciar en la figura 6.1.

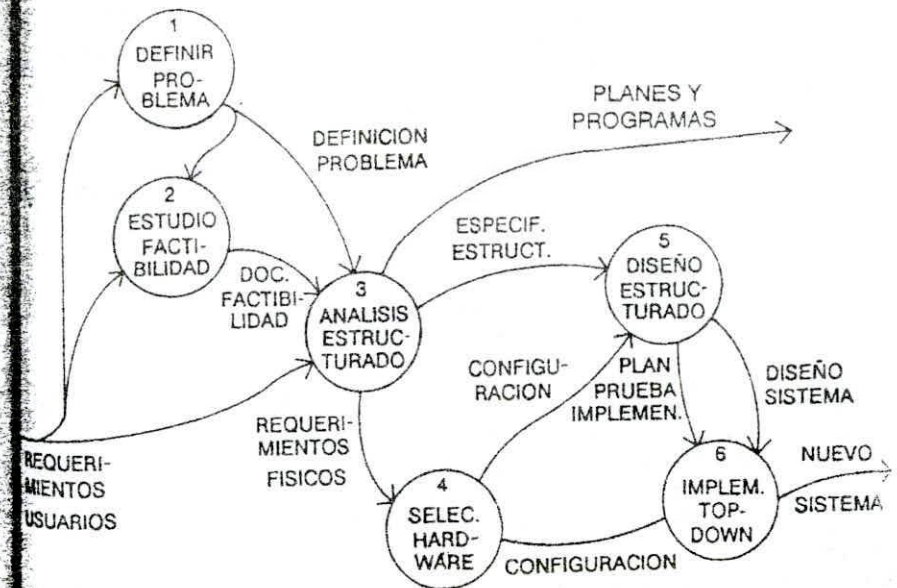


Figura 6.1: Desarrollo de Sistemas Informáticos

Dentro del desarrollo estructurado de sistemas, la fase de análisis requiere de un conjunto de subfases que es necesario llevar a cabo a objeto de cumplir con su objetivo específico, que consiste en lograr una especificación del sistema.

La figura 6.2 muestra aquellas subfases y los documentos que las relacionan. Todas ellas forman parte de las tareas que se debe desarrollar durante el análisis, en cuanto etapa del desarrollo de un sistema. Estas tareas tienen como marco de referencia la definición del problema y, si se ha realizado, el estudio de factibilidad, y como punto de partida la función o proceso organizacional objeto de estudio informático.

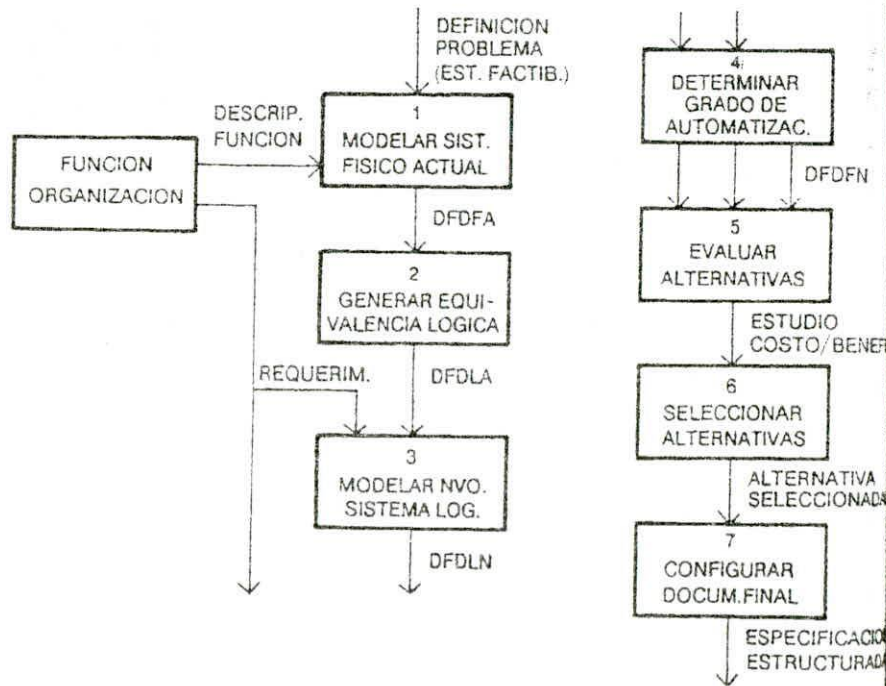


Figura 6.2: Subfases del Análisis

Su resultado es una especificación que describe el nuevo sistema, destacando aquellas subfunciones que se llevarán a cabo automatizadamente de las que se realizarán de modo manual, explicitando para estas últimas los procedimientos que regirán su ejecución. Destaca por lo tanto, como un aspecto fundamental, los requerimientos de tratamiento automatizado que la función precisa.

Las tareas del análisis se pueden dividir en siete subfases, las cuales, representadas en términos de la estructura de los modelos del tipo grafo -que son aquellos que se utilizan para la representación de sistema-, asumen la conformación que muestra la figura 6.3.

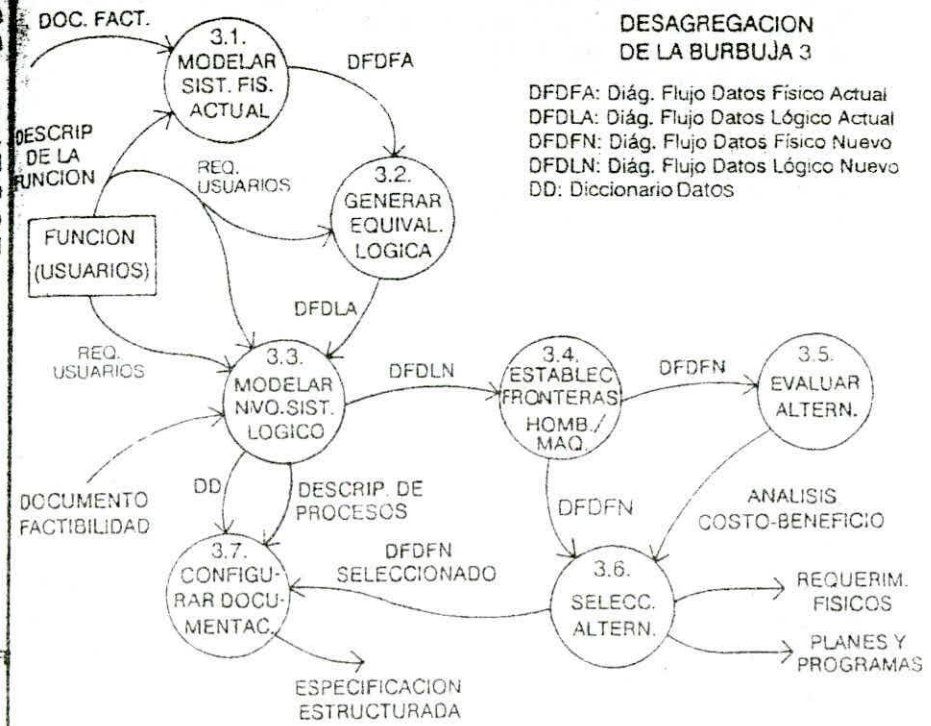


Figura 6.3: Subfases del Análisis

Los alcances de estas subfases son:

- La función, corresponde al proceso organizacional o subsistema sometido a tratamiento informático, y que fue ya definido como proyecto de desarrollo en la fase de definición del problema. Es aquella que se someterá a análisis.
- Modelar sistema físico actual es la primera subfase del análisis y provee como producto principal un diagrama de flujo de datos físico actual, (DFDA), que describe los subprocesos de la función y sus relaciones en términos de referentes físicos.
- Generar equivalente lógico del esquema físico actual, es una segunda subfase, de la cual resulta un diagrama de flujo de datos lógico actual (DFDLA) que describe las mismas subfunciones, pero ahora desde un punto de vista del rol orgánico-funcional que les corresponde como componentes del sistema.
- La tercera subfase, modelar el nuevo sistema lógico, genera como resultante principal un diagrama de flujo de datos lógico nuevo (DFDLN), el cual muestra las modificaciones incorporadas a las subfunciones orgánicas. Se describe en esta subfase los componentes del sistema redefinido y sus interfaces, y se documenta el contenido de los flujos de datos que relacionan aquellos componentes y los procedimientos en virtud de los cuales se transforman, al interior del sistema, los diferentes flujos de datos.

- Determinar el grado de automatización, es la cuarta subfase, y le corresponde la determinación de aquellas subfunciones que serán automatizadas, generando un conjunto de alternativas diferenciadas justamente, por el grado de automatización que incorporan.

Los modelos así generados se conocen como diagramas de flujo de datos físicos nuevos (DFDFN).

- La quinta subfase consiste en evaluar cada alternativa, es en donde se determinan las inversiones, costos y beneficios asociados a cada una de las alternativas generadas en la fase anterior, evaluándolas mediante la utilización de las herramientas tradicionales de evaluación de proyectos, tales como el valor actual neto (VAN), la tasa interna de retorno (TIR), etc. Como producto de esta subfase se emite un documento denominado estudio costo/beneficio.
- Seleccionar alternativa, es la sexta subfase y su tarea fundamental consiste en la elección de la mejor opción dentro de aquellas que fueron evaluadas en la subfase anterior. Su resultado es una alternativa seleccionada.
- La última subfase, configurar documentación final, consiste en dar la coherencia necesaria al documento que detalla el trabajo de la fase de análisis, es decir, la especificación estructurada.

6.1. Modelar sistema físico actual

El modelado del sistema físico actual es la primera subfase dentro del conjunto de tareas que corresponden al análisis estructurado, y su propósito es llevar a cabo una descripción exhaustiva de la función o proceso bajo estudio, para lo cual dicha función se desagrega en sus componentes básicos, conformando un modelo que refleja la forma actual en que la función o proceso sometido a estudio, lleva a cabo sus actividades. Se trata de lograr una visión comprensiva de la forma en que se llevan a cabo las tareas actualmente documentando lo observado, y fundamentalmente en términos de la conformación del modelo que describe el sistema.

Para poder conformar tal modelo es preciso tener como punto de partida algunos antecedentes previos. Estos son:

- Una visión global del sistema.
- El ámbito del proyecto.
- La identificación de los agentes del proceso que están involucrados en las diferentes actividades que debe llevar a cabo.

lo específico, la tarea consiste en reflejar en un modelo del tipo grafo las distintas subfunciones y sus relaciones, tal como actualmente desarrollan sus actividades.

Las interfases corresponden a flujos de datos y los procesos a las transformaciones a las cuales dichos flujos de datos se someten. En estos diagramas las operaciones se registran desde el punto de vista de los datos, entendiéndolos como instancias que fluyen desde o hacia algunas subfunciones que los transforman. Lo que se modela en consecuencia, es un diagrama de flujo de datos.

El resultado final es un documento denominado diagrama de flujo de datos físico actual y que, según muestra la figura 6.2, es aquel con el cual se debe llevar a cabo la segunda subfase.

Dado que esta subfase corresponde a un primer acercamiento al sistema real, lo que se modela es el reflejo de lo que las personas involucradas en el quehacer de las diferentes subfunciones habitualmente identifican como elementos constitutivos del sistema, es decir, el diagrama debe reflejar la percepción particular que ellos tienen respecto de las operaciones que realizan.

Esto implica el uso de su propia terminología, la utilización de su propio esquema de división de tareas, los nombres reales con que se identifican los documentos y la forma en que se reconocen las diferentes subfunciones, aunque se expresen en términos de nombres de personas o departamentos.

Es decir, en los diagramas los nodos se identifican ya sea con nombres de personas, departamentos o secciones y los flujos de datos con los nombres o señas con que se reconoce habitualmente los documentos. Incluso, por ejemplo, un flujo de datos puede identificarse por el color, formato o código del documento en el cual reside.

La razón fundamental para proceder de este modo en este primer acercamiento, es contar con un instrumento riguroso -el modelo- y eficazmente explícito -la terminología- que permita primero la comprensión, y luego la validación y aceptación del sistema en cuanto reflejo exacto de la realidad que se está analizando.

En consecuencia los pasos a seguir para completar esta primera tarea son:

- Determinar la función a estudiar.
- Identificar los agentes involucrados en el proceso; es decir, las personas comprometidas en las diferentes actividades que lleva a cabo.
- Configurar el modelo -tipo grafo- mediante entrevistas con las personas y observación directa.
- Modelar el diagrama de flujo de datos físico.

- Presentar el modelo al usuario para revisar y corregir la versión obtenida.
- Generar el modelo exacto del sistema denominado diagrama de flujo de datos físico actual.
- Documentar el trabajo acompañándolo con muestras de los documentos que contienen los diferentes flujos de datos y archivos considerados.

Un ejemplo de un diagrama de flujo de datos físico actual es el que muestra la figura 6.4. El modelo configurado corresponde a un proceso de determinación del costo total de un departamento de producción que lleva a cabo la fabricación a pedido de una máquina o herramienta.

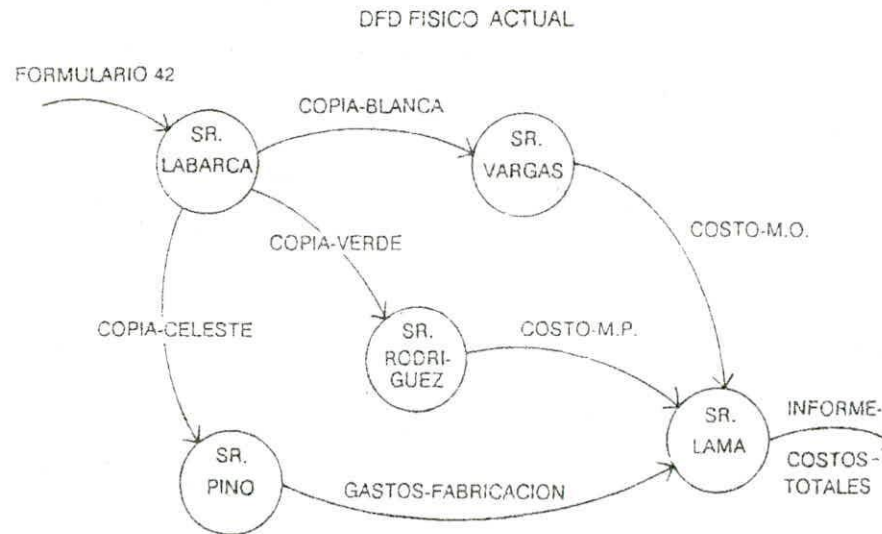


Figura 6.4: DFDFA

El proyecto es primero determinado técnicamente, especificando en un documento las horas-hombre requeridas, las materias primas y los gastos generales en que se estima se incurrirá. El documento de especificación se emite con tres copias, una relativa a las materias primas, otra a los costos de mano de obra y la otra a los gastos de fabricación. Cada una de ellas se somete por separado a estudios para determinar los costos respectivos de tal modo de poder posteriormente determinar los costos totales.

6.2. Generar equivalencia lógica

La subfase anterior genera un diagrama de flujo de datos físico actual, a esta subfase le corresponde ahora identificar en aquel diagrama, el modelo lógico subyacente en virtud del cual el proceso lleva a cabo su misión. Para ello se eliminan del diagrama anterior todos los referentes físicos y se sustituyen por referentes lógicos, es decir, por expresiones que reflejen conceptualmente cada función o proceso que lleva a cabo una cierta transformación de un flujo de datos de entrada en flujo de datos de salida, tal como ocurre con la sustitución del flujo de datos formulario 4-2 por el flujo de datos "requerimientos de trabajo".

La razón por la cual se incluyen referentes físicos en el modelo previo, es que ellos facilitan la comunicación con las personas que llevan a cabo las tareas correspondientes al proceso, ya que ellos son los hitos naturales de su ambiente de trabajo.

Los referentes físicos les facilitan la comprensión del modelo que representa el sistema, y por ende, su reconocimiento y posterior validación. Sin embargo, obviamente esta representación, además de conformarse en términos de elementos de carácter no permanente -por lo cual está expuesto a la pérdida de vigencia-, no refleja conceptualmente lo que es la función en su esencia como componente organizacional.

Por ello, en consecuencia, se hace necesario transformar el diagrama de flujo de datos lógico actual en lo que en la figura 6.3 se identifica como diagrama de flujo de datos lógico actual.

La figura 6.5 muestra el equivalente lógico del diagrama físico de la figura 6.4.

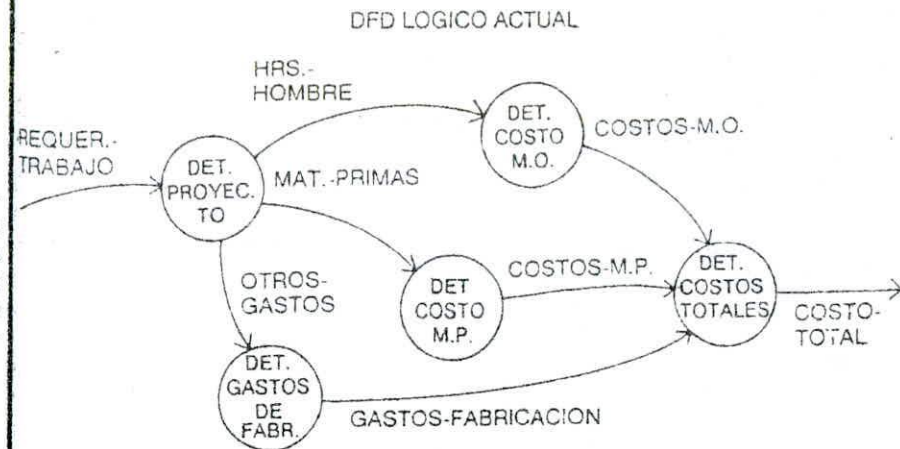


Figura 6.5: DFDLA

6.3. Modelar nuevo sistema lógico

El modelo representado en términos de un diagrama de flujo de datos lógico actual, corresponde a la forma en que actualmente se llevan a cabo las tareas en la función o proceso objeto de análisis. Sin embargo, la dinámica organizacional exige normalmente de ciertas redefiniciones, las cuales deben necesariamente abordarse. En lo específico, se debe incluir aquellos cambios estructurales que se han especificado para la función en la definición del problema y estudio de factibilidad, o que se perciben en las entrevistas con personas involucradas en las tareas cotidianas de la función. Estas modificaciones pueden consistir en la incorporación, en la eliminación, en la conjunción o separación de subfunciones.

Como resultado de ello, se presenta una descripción lógica redefinida de la función objeto de estudio que señala el qué hacer pero aún no el cómo hacer, en otras palabras no se especifica si alguna o algunas de las subfunciones serán manuales o automatizadas. El modelo resultante describe las relaciones entre las partes componentes -las subfunciones-, y se complementa con las reglas que rigen las acciones que deben llevar a cabo, y que corresponden a las especificaciones de procedimientos. También con las descripciones de los flujos de datos y archivos que relacionan aquellas subfunciones.

6.3.1. Construcción de un modelo lógico para un sistema futuro

La transformación del modelo lógico actual en un modelo lógico nuevo corresponde, en gran medida, a lo esencial de la etapa de análisis en el desarrollo de un sistema. Es en esta subfase donde se construye el modelo lógico del nuevo sistema. Hasta el momento, se ha dedicado todos los esfuerzos al estudio de las actuales operaciones, es decir, a la forma en que la función o proceso objeto de análisis lleva a cabo su actividad en la actualidad. Sin embargo, una vez completado este estudio es necesario enfrentar el problema relativo a la forma de operación deseada para la función.

Para el desarrollo de esta subfase, es fundamental el Documento del Cambio. Este documento es un producto de la fase de definición del problema y presenta una exposición de lo que se desea del nuevo sistema. En términos generales, se refiere ya sea a la automatización de un conjunto de procedimientos, a la centralización o descentralización de algunas subfunciones, a la sustitución de algún procedimiento "batch" por un trabajo en línea o a la incorporación de un nuevo conjunto de actividades.

Para la construcción de un modelo lógico del nuevo sistema se cuenta con el modelo lógico del sistema actual, documentado en términos de diagramas de flujo de datos, e incluso a veces incluyendo diccionario de datos y especificaciones

de procedimientos. Cuando la transformación del modelo lógico actual en un modelo lógico nuevo esté terminada, se tendrá un modelo lógico de características similares al que describe el sistema antiguo. En ambos casos, se solicita un conjunto interconectado de subfunciones sin distinguir entre procedimientos automatizados y manuales.

La figura 6.6 muestra los elementos involucrados en esta subfase. El modelado del nuevo sistema lógico parte con la identificación del Dominio del Cambio en el diagrama de flujo de datos lógico actual, y con la consecuente conservación de aquellas subfunciones que no sufrirán modificaciones.

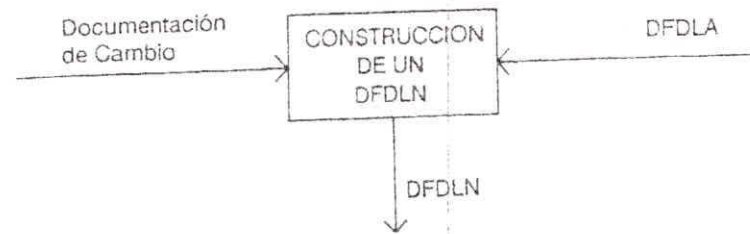


Figura 6.6.: Construcción de un DFD lógico nuevo.

6.3.2. El Dominio del Cambio

Si el analista ha sido lo suficientemente cuidadoso en la selección del contexto para el estudio del sistema actual, incluirá muchos más aspectos que los que corresponden al área susceptible de ser afectada por el cambio especificado en la Definición del problema. Por esta razón, una parte del nuevo modelo lógico será exactamente igual a la parte correspondiente del antiguo modelo lógico.

En relación al diagrama de flujo de datos lógico actual, el Dominio del Cambio se define como el conjunto de todos los componentes que son susceptibles de ser cambiados, frente a una nueva forma de concebir la función objeto de análisis. El Dominio del Cambio es la parte del antiguo modelo lógico que será modificada. El complemento del Dominio del Cambio es la parte que se puede conservar.

Para establecer el Dominio del cambio, se revisa cada burbuja del diagrama de flujo de datos lógico actual, trabajando en el nivel de mayor desagregación. Para cada burbuja del último nivel, se debe observar si hay algo de este proceso que pudiera ser modificado en relación a la forma de ser actual, es decir, si podría ser automatizado, automatizado en parte, eliminado o simplemente hecho de una manera diferente a la actual. Si es así, en otras palabras, si el proceso no es

afectado por cambio alguno, entonces queda fuera del Dominio del Cambio. De esta forma se puede separar el diagrama de flujo de datos lógico actual en dos dominios: el afectado y el no afectado por el cambio.

Una vez que se ha identificado el Dominio del Cambio, el diagrama de flujo de datos aparecerá marcado, tal como lo muestra la figura 6.7.

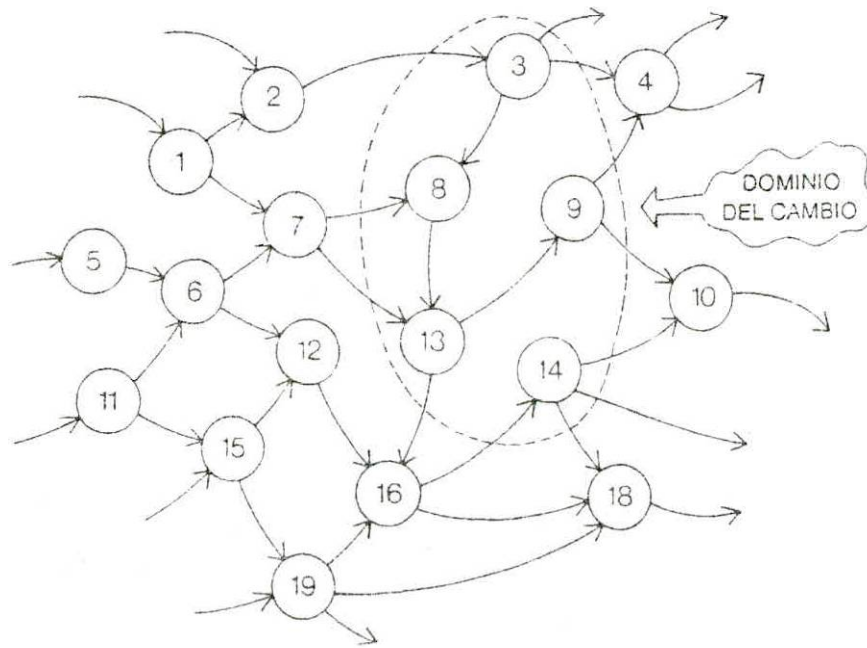


Figura 6.7: Dominio del Cambio

Puede ocurrir que exista en el diagrama de flujo de datos más de un área del Dominio del Cambio completamente desconectadas. Esto es sólo una consecuencia de la forma en que el diagrama de flujo de datos está dispuesto en el papel, lo que no tiene mayor importancia.

Una vez identificado el Dominio del Cambio, todas las burbujas que lo configuran se reemplazarán por una sola burbuja, denominada justamente, Dominio del Cambio, a la cual se conectan las interfaces que a ella llegan, tal como lo muestra la figura 6.8.

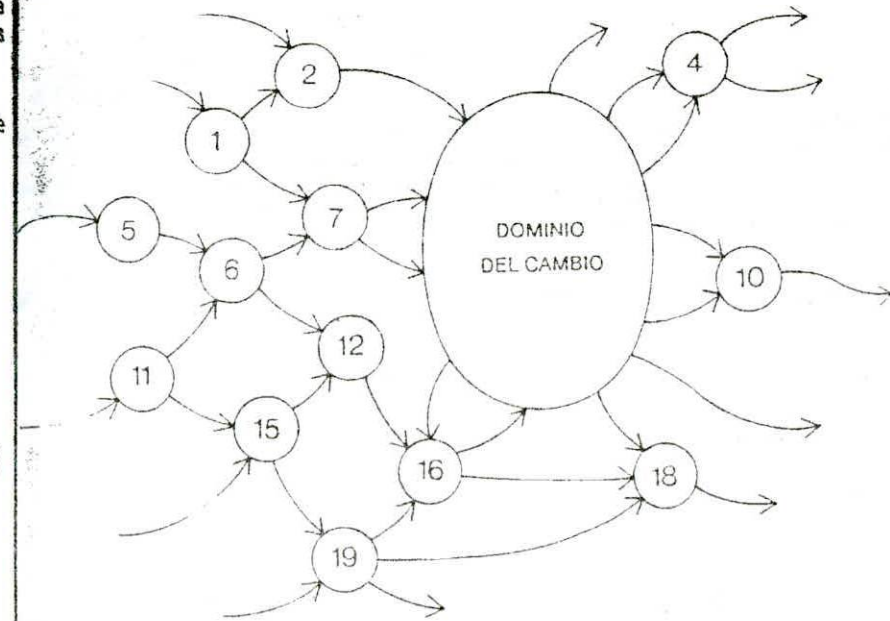


Figura 6.8: Dominio del Cambio

El diagrama de flujo de datos resultante muestra el Dominio del Cambio sólo con las interfaces de composición conocida, las cuales están definidas en cuanto a contenido y organización en el Diccionario de Datos. Por definición, todo lo que es desconocido, es decir, lo que aún debe ser especificado, está dentro de los límites del Dominio del Cambio, todo lo que se encuentra fuera es conocido.

Cabe destacar que al interior del Dominio del Cambio no sólo está el nuevo sistema computacional. También es posible que hayan algunos procesos manuales nuevos y otros simplemente modificados, en otras palabras, lo que se ha hecho es delimitar el problema.

Cuando se debe definir el contexto de la función objeto de análisis, es conveniente no restringirlo indebidamente, sino más bien ampliarlo lo suficiente como para tener la certeza que no se ha dejado fuera ningún componente crítico para el sistema. Esto significa que el contexto original, objeto de análisis, debe incluir las subfunciones necesarias para poder definir adecuadamente el Dominio del Cambio, y las burbujas asociadas con las cuales aquel se relaciona a través de flujos de datos de composición conocida. Es decir, debe ser posible aislar todo el Dominio del Cambio dentro del modelo lógico actual. Dicho de otra forma, el contexto original de estudio debe ser lo bastante amplio de manera que todos los cambios queden dentro de su contexto, delimitados por un conjunto de

6.3.3. Partición del Dominio del Cambio

Una vez que se tiene el dominio delimitado, la generación del nuevo modo consiste en el particionamiento de arriba hacia abajo de la burbuja que representa el Dominio del Cambio en el diagrama de flujo de datos.

Este particionamiento difiere de aquellos realizados en las subfases anteriores del análisis por varias razones:

- † En primer lugar se realiza desde arriba hacia abajo, ya que lo único claramente establecido es el nivel más global, es decir, el Dominio del Cambio y sus entradas y salidas, en cuyo contexto las subfunciones componentes aún no están establecidas. Esta es la razón por la cual no es posible comenzar con la descripción de los niveles inferiores como ocurre con la formulación de los diagramas de flujo de datos físicos y lógicos actuales.
- † El particionamiento está determinado por los estándares del análisis y no por los requerimientos del usuario. De este modo se particiona el Dominio del Cambio de acuerdo principalmente con la necesidad de minimizar las interfaces y alcanzar para cada subfunción niveles adecuados de cohesión.
- † El Documento del Cambio es aquel que determina todas las modificaciones que han de introducirse a la función sometida a análisis, y que se traduce en subfunciones agregadas o eliminadas, en nuevos flujos de datos, etc.

En otras palabras, es en esta instancia en la cual se configura el nuevo sistema. Cabe destacar que el Análisis Estructurado no provee herramientas para esta tarea, pero sí para la documentación de aquella nueva concepción del sistema.

La construcción de un modelo lógico de cualquier nuevo sistema, incluye los diagramas de flujo de datos jerárquicos, un Diccionario de Datos y la creación de un conjunto integrado de especificaciones de procesos. Para esta tarea el Análisis Estructurado dispone de un conjunto de herramientas y criterios de trabajo entre los cuales es conveniente destacar:

- I) Considerar como documento de trabajo el diagrama de flujo de datos lógico actual y el documento del cambio.
- II) Establecer niveles adecuados de cohesión y acoplamiento. En cada nivel se debe hacer todo lo posible para minimizar las transferencias de datos entre procesos y archivos. Si se combina dos procesos y el resultado es un sistema menos complejo que aquel en el cual ambos están separados, quiere decir que se ha mejorado el particionamiento. Por su parte, si se divide un proceso y cada uno de los nuevos subprocesos maneja ahora un conjunto menos complejo de flujos de datos en su interior, como en su relación con sus otros subprocesos, nuevamente se ha mejorado el particionamiento.

La definición de un diagrama de flujo de datos lógico nuevo, equivalente a la redefinición del diagrama de flujo de datos lógico actual, requiere de una exhaustiva revisión de la manera en que actualmente la función lleva a cabo sus tareas.

Esto implica someter a revisión las subfunciones que la conforman, a objeto de observar si existe carencia de alguna de ellas, si otra u otras no cumplen un rol significativo y necesario, o si existen algunas que realizan deficientemente su cometido.

La definición de un nuevo diagrama de flujo de datos lógico implica también eliminar flujos de datos innecesarios, o incorporar otros, a la par que incorporar o eliminar instancias de almacenamiento de datos o archivos.

Estas redefiniciones a que se somete la función o proceso enfrentada a tratamiento informático, descansan fundamentalmente en dos criterios básicos: la cohesión y el acoplamiento. La cohesión expresa la medida en la cual todas las tareas de transformación o proceso se corresponden entre sí. El acoplamiento se refiere al nivel de interacciones que se producen entre las subfunciones.

En estos términos, la efectividad del quehacer de la función estará en gran medida asociado a la conformación estructural que asumen las subfunciones que la componen, es decir, a los niveles de cohesión y acoplamiento. Una adecuada conformación estructural se orienta preferentemente hacia un bajo acoplamiento y una alta cohesión.

Un sistema altamente cohesionado es aquel en el cual todas las partes contribuyen a una sola función.

Un bajo acoplamiento es aquel en el cual los flujos de datos que conectan dos subfunciones corresponden a los mínimos indispensables.

Prestar atención a los nombres. Un flujo de datos debe tener una cierta integridad conceptual y ésta debe reflejarse en el nombre de manera natural. En otras palabras, el nombre debe ser naturalmente representativo del conjunto de datos que configura el flujo. Del mismo modo, los procesos deben nominarse de acuerdo a la función que realizan, considerando que lo óptimo es lograr en el nivel inferior una función bien definida por cada proceso susceptible de nominar, también de manera natural y directa, justamente a partir de dicha función. No hay que olvidar que el uso de las herramientas del Análisis Estructurado recomienda no usar palabras tales como proceso, manipulación, datos, entrada o salida. También es conveniente recordar que si un proceso o flujo de datos es difícil de nominar, ello es signo de un cohesionamiento inadecuado que es conveniente revisar.

Limitar la desagregación. Los diferentes modos de desagregación de un diagrama de flujo de datos, no deben contener un excesivo número de

burbujas. Estudios empíricos demuestran que no es aconsejable que ellas contengan más de siete procesos. En realidad se sugiere 7 ± 2 burbujas por diagrama [MIL-56].

- V) Verificar la corrección de los flujos de datos. Es preciso asegurar que cada uno de los flujos de datos esté especificado y que además, todos los diagramas en un cierto nivel estén balanceados respecto de sus diagramas padres.
- VI) Respetar la regla de conservación de datos. Cualquier elemento de datos que fluya fuera de un proceso debe haber ingresado antes de alguna forma, a menos que se trate de un parámetro inherente al proceso. Si hay algún error significa que alguna interfase ha sido omitida.
- VII) Mantener el Diccionario de Datos actualizado. Siempre que se crea un nuevo flujo de datos, se debe definir de inmediato en el Diccionario de Datos.

6.3.4. Prueba de la nueva especificación lógica

El diagrama de flujo de datos lógico nuevo sólo estará completo cuando se tenga un conjunto coherente y equilibrado de diagramas de flujo de datos con toda la documentación de apoyo, es decir, con todas las descripciones y con todas las especificaciones de procedimientos para los procesos del último nivel del diagrama jerárquico de flujo de datos.

6.4. Determinar grado de automatización

Una vez establecido el conjunto de subfunciones definitivo del sistema a construir, es decir, una vez determinado el qué hacer, es preciso especificar cuáles de aquellas subfunciones se llevarán a cabo automatizadamente y cuales de modo manual.

En otras palabras, es preciso introducir modificaciones al diagrama de flujo de datos lógico nuevo, relativas a aspectos que podrían denominarse de corte físico. Corresponde a la incorporación de niveles de automatización, por lo cual no se afecta la estructura funcional lógica del sistema, sino la forma específica en que se llevarán a cabo las tareas en cada subfunción.

Una vez establecido aquello que será manual y aquello que será automatizado, se genera un diagrama de flujo de datos físico nuevo, en el cual estas formas

de llevar a cabo las tareas se hacen distinguibles mediante un límite denominado frontera hombre-máquina.

En otras palabras, esta fase consiste en introducir modificaciones al diagrama de flujo de datos lógico nuevo, que específicamente corresponden a la incorporación de ciertos niveles de procesamiento automatizado en algunas o en todas las subfunciones. Cabe reiterar que con ello no se modifica la estructura funcional lógica del sistema, puesto que sólo se producen ciertos cambios en relación al cómo hacer de la función.

Cabe señalar que a este nivel solamente se establece cuáles subfunciones se llevarán a cabo automatizadamente, sin detallar la forma específica en que ello ocurrirá.

Al sugerir la incorporación de grados de automatización a un sistema, obviamente existen diversas opciones que pueden ir desde la automatización total, hasta una concepción absolutamente manual del sistema.

En consecuencia, esta subfase del análisis está destinada a la incorporación de algunos referentes físicos al modelo lógico de la subfase anterior, el cual no distingue entre funciones automatizadas y manuales. Cabe destacar que todo proyecto informático exige de la fase de análisis la determinación del ámbito del esfuerzo de desarrollo que se debe enfrentar a continuación. Sólo cuando se ha establecido qué procesos elementales del modelo serán realizados automatizadamente y cuales de modo manual, se habrá completado la especificación que requiere al diseño, la siguiente fase en el desarrollo del sistema.

6.4.1. Lo automatizado y lo manual en la especificación del sistema

La fase de análisis no se considera concluida sino hasta que se establezca y documente la interfase hombre-máquina.

La Especificación Estructurada, que describe el sistema, debe hacer evidente esta interfase y ayudar a resolver los posibles problemas que se presenten.

Existen dos formas de incorporar la información acerca de la interfase hombre-máquina dentro de la Especificación Estructurada. La primera es documentarla, usando para ello un diagrama de flujo de datos, y la segunda integrar lo automatizado como un todo en el primer nivel del diagrama de flujo de datos.

6.4.2. Especificación de lo automatizado mediante un diagrama de flujo de datos adicional

La figura 6.9 muestra la primera alternativa, el uso de un diagrama de flujo de datos para documentar las relaciones hombre-máquina. Esta relación, se expresa en términos de los flujos de datos que se mueven entre los procedimientos manuales y automatizados.

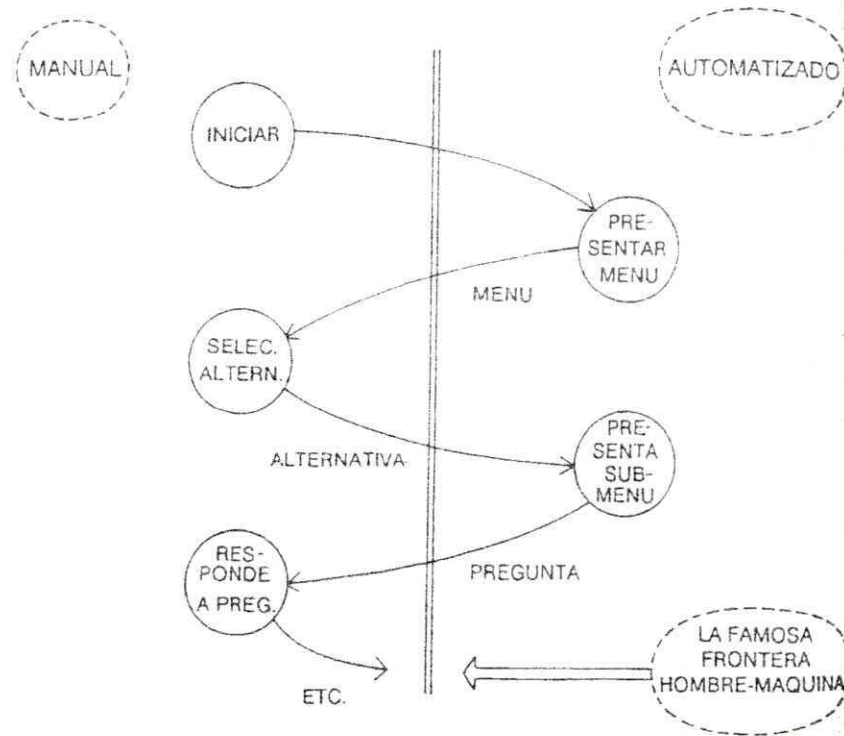


Figura 6.9: Documentación Interfase H/M mediante DFD

De acuerdo a este proceder, ahora la Especificación Estructurada consistirá de un conjunto de diagramas jerárquicos de flujo de datos, de un diccionario de datos, de la especificación de procedimientos para las burbujas elementales y además de los diagramas de flujo de datos que explicitan la relación hombre-máquina. La dificultad que trae consigo esta forma de documentar es que los procesos que están en la frontera se encuentran descritos en dos partes, por lo cual cualquier cambio que se desee incorporar ya sea a la frontera misma o a los procesos que lindan con ella, requerirá de una actualización redundante de la Especificación Estructurada.

6.4.3. Especificación de lo automatizado mediante redefinición del diagrama de flujo de datos

La figura 6.10 muestra la interfase hombre-máquina documentada de manera diferente. En este caso, se logra el objetivo sin crear un documento de interfase, sino mediante la reagrupación del diagrama de flujo de datos para explicitar la interfase hombre-máquina.

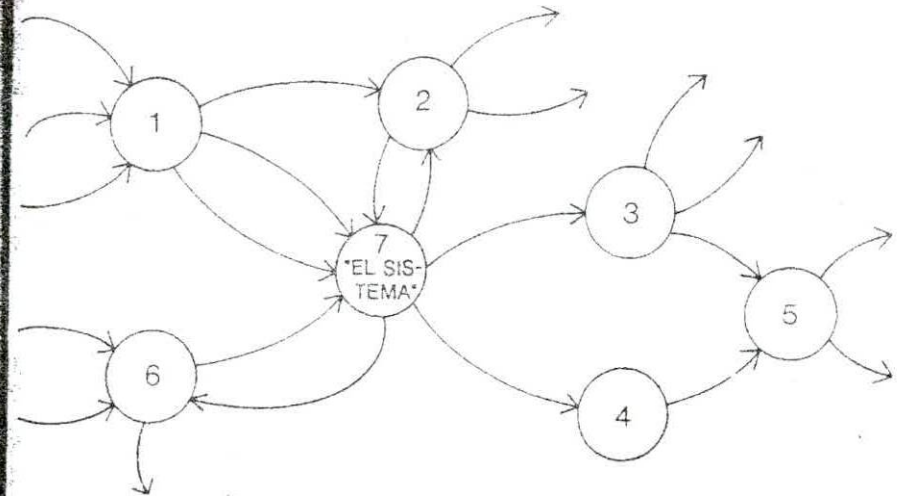


Figura 6.10: Documentación interfase H/M mediante reagrupación

Específicamente, en este caso, toda interrelación con lo automatizado se concentra en el nivel de mayor agregación del diagrama.

En la figura 6.10 se muestra el sistema automatizado como una sola burbuja, la burbuja 7, por lo tanto la frontera entre lo automatizado y lo manual está alrededor de ella. Corresponde, de este modo, a un conjunto de flujos de datos que entran y salen de la burbuja 7. Cabe destacar que, de acuerdo con el balanceo de mallas, no puede haber interfases adicionales entre los procesos automatizados y los procedimientos manuales en los niveles más bajos del diagrama.

De esta forma entonces, todos los diagramas derivados de la burbuja 7, la información respecto de los flujos de datos detallada en el diccionario de datos y las especificaciones de procedimientos asociadas a los procesos de último nivel describen la función automatizada del sistema.

Respecto de las burbujas 1 a la 6, todos sus niveles inferiores y la documentación de apoyo, describen los procedimientos manuales.

En consecuencia, la Especificación Estructurada está compuesta tanto por los aspectos manuales como por los aspectos automatizados del sistema, utilizando en este caso las mismas herramientas para describir ambas dimensiones.

6.4.4. Documentación de los procedimientos manuales

El Análisis Estructurado sugiere que los procedimientos relativos al funcionamiento del sistema, en lo que se refiere a sus aspectos no automatizados se completan y documentan durante la fase de análisis.

Esta es una diferencia fundamental con el enfoque clásico, en el cual los procedimientos para el uso del sistema no se completan sino en las etapas posteriores del proyecto, es decir al momento de conformar los manuales del usuario.

La razón para incluir durante la fase de análisis, la descripción de los procedimientos manuales radica en que el usuario está normalmente en desventaja en cualquier discusión que tenga relación con procedimientos automatizados. Para él lo único importante es cómo hará uso del sistema automatizado. Por lo tanto es evidente que se logrará de él un mejor aporte cuando las decisiones se centran en los aspectos sobre los cuales tiene mayor dominio, como lo son los manuales del sistema.

Por último como señala Tom De Marco [DEM-79]:

- El trabajo debe hacerse de todos modos, y haciéndolo más temprano se logra mejorar y hacer más fácil el resultado final.
- La documentación de los procedimientos durante la fase de análisis ayuda a privilegiar una participación más significativa del usuario.
- En la medida en que el usuario esté más al tanto de cómo el sistema afectará su propia actividad, resultará más fácil obtener su aceptación, eliminando una serie de pruebas paralelas. De esta forma también el resultado será menos sorpresivo para él.

Todos estos aspectos manuales del sistema se explicitan en la Especificación Estructurada. Es decir aquellas subfunciones que no serán automatizadas y que aparecen en el diagrama pero deben describirse mediante los diagramas asociados a las subfunciones derivadas, mediante el diccionario de datos y a través de la especificación de procedimientos cuando la desagregación alcanza a los subprocesos de último nivel.

4.5. La generación de alternativas de automatización

Cabe destacar que, metodológicamente, se sugiere proponer más de una alternativa, cada una de las cuales incorpora diferentes grados de automatización a la función. Cada una ellas se somete posteriormente a un estudio costo/beneficio, en virtud del cual se seleccionará, obviamente, la más ventajosa.

La transformación del modelo lógico del nuevo sistema en un modelo físico requiere llevar a cabo dos tareas fundamentales. Estas son:

- Proponer un conjunto de alternativas y establecer, para cada una, un cierto grado de automatización.
- Incorporar subfunciones según las facilidades de la configuración.

La figura 6.11 representa el diagrama de flujo de datos de un sistema, representado en su último nivel de desagregación.

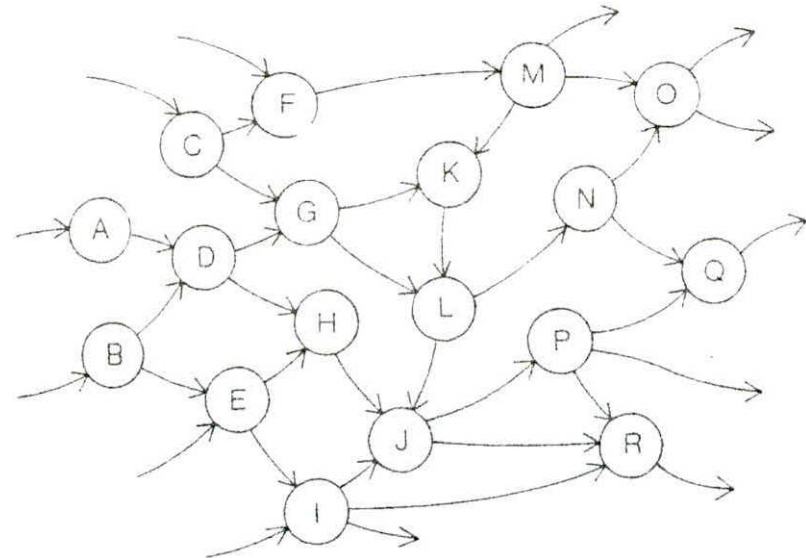


Figura 6.11: DFD a último nivel

Las figuras 6.12 y 6.13, por su parte, representan dos alternativas planteadas como parte de la transformación del diagrama de flujo de datos lógico en diagrama de flujo de datos físico. Cabe reiterar que estos diagramas de flujo de

datos difieren sólo en el grado de automatización. Si se observa la figura 6.12 se puede afirmar que corresponde a una solución de automatización casi total porque prácticamente todos los subprocesos están incluidos en la zona automatizada. En una primera aproximación se puede decir que esta alternativa será más cara que la representada en la figura 6.13, pero obviamente también tendrá algunos beneficios adicionales.

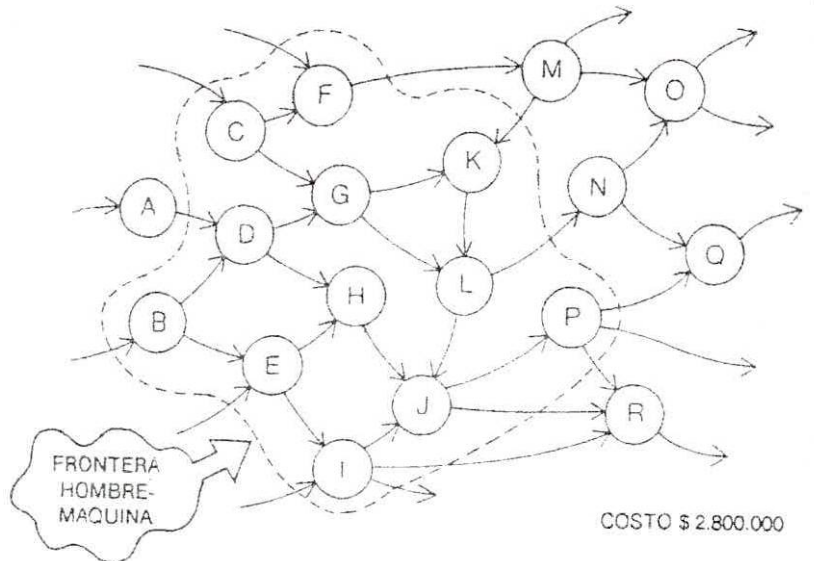


Figura 6.12: DFD alternativo

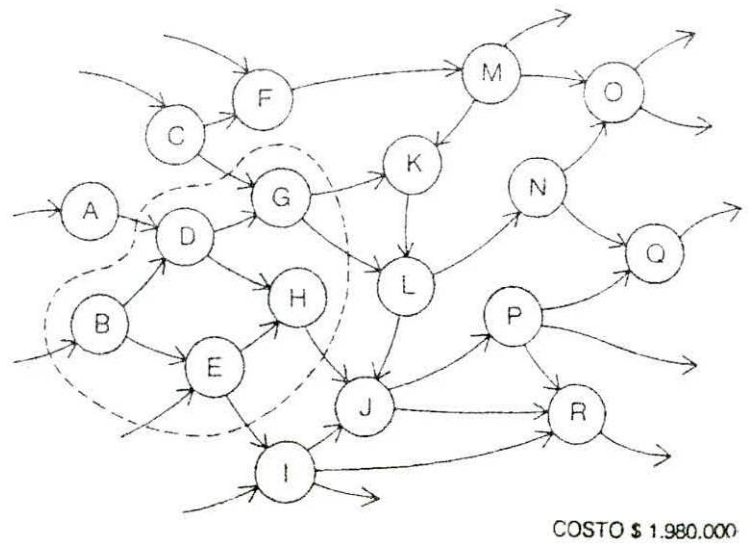


Figura 6.13: DFD alternativo

Se debe destacar que, en cualquier caso, las soluciones con mayor nivel de automatización requieren de configuraciones y de software más sofisticado.

Además, cuando se plantea el objetivo de establecer algunas alternativas de solución, en función del grado de automatización, se infiere que cada una de ellas debe ser aceptable (por definición). Obviamente sólo una será seleccionada, pero ninguna de ellas puede ser inaceptable a priori, y si así fuese, ésta no debería considerarse como una alternativa viable.

Existe una tendencia generalizada que es preciso evitar, ésta es la tentación de seleccionar una alternativa considerándola a priori como la única opción. Esto significa que, invariablemente, el sistema será siempre el sistema del analista. Si los ejecutivos y usuarios no participan en el proceso decisional para determinar la magnitud del sistema a construir, entonces el carácter del sistema, en cuanto al grado de automatización que incorpora, será de exclusiva responsabilidad del analista.

Para obviar esta situación es conveniente presentar un conjunto de alternativas viables y, más aún, que los ejecutivos y usuarios participen en la selección. De este modo la responsabilidad por el efecto del sistema construido es compartida por todos. Sólo así se puede afirmar que el sistema responde a las expectativas de la mayoría. El compartir responsabilidades es esencial para el término exitoso del proyecto.

6.4.6. Incorporación de subfunciones según las facilidades de la configuración

Se puede decir que, en general, una solución más cara normalmente involucra un grado de automatización mayor. Incluso ocurre normalmente que una solución de este tipo, en términos de hardware y software requerido, permite automatizar subfunciones que con las más económicas no es posible lograr, ya que existen ciertas subfunciones que dependen de las características de la configuración. Esto significa que es posible incorporar subprocesos que sólo se pueden llevar a cabo debido a las facilidades que provee la configuración disponible. En estos términos entonces, cada una de las alternativas podría incorporar una o más subfunciones dependientes de las facilidades de dicha configuración. Estas subfunciones pueden, por ejemplo, referirse a la generación de datos con ciertos grados de sofisticación que de otro modo no podrían obtenerse.

La consecuencia directa de estas incorporaciones es la redefinición del modelo lógico, es decir, del diagrama de flujo de datos lógico nuevo y, consecuentemente la obtención del diagrama de flujo de datos físico nuevo correspondiente. El resultado final, como se señaló con anterioridad, muestra varios modelos o diagramas de flujo de datos físicos nuevos, uno por alternativa. Cabe destacar que cada modelo sigue siendo esencialmente lógico, ya que lo único de carácter físico que se manifiesta es la determinación de las fronteras hombre-máquina.

Cuando se identifican aquellas subfunciones, dependientes de la configuración, se debe conformar un diagrama de flujo de datos tal como aparece en la figura 6.14, en donde los nuevos subprocesos incorporados aparecen ennegrecidos, debiendo documentarse tal como ocurre con las otras subfunciones, es decir, especificando en el diccionario de datos sus flujos de datos de entrada y salida y los procedimientos que han de regir las transformaciones a que se someten los primeros para generar los segundos.

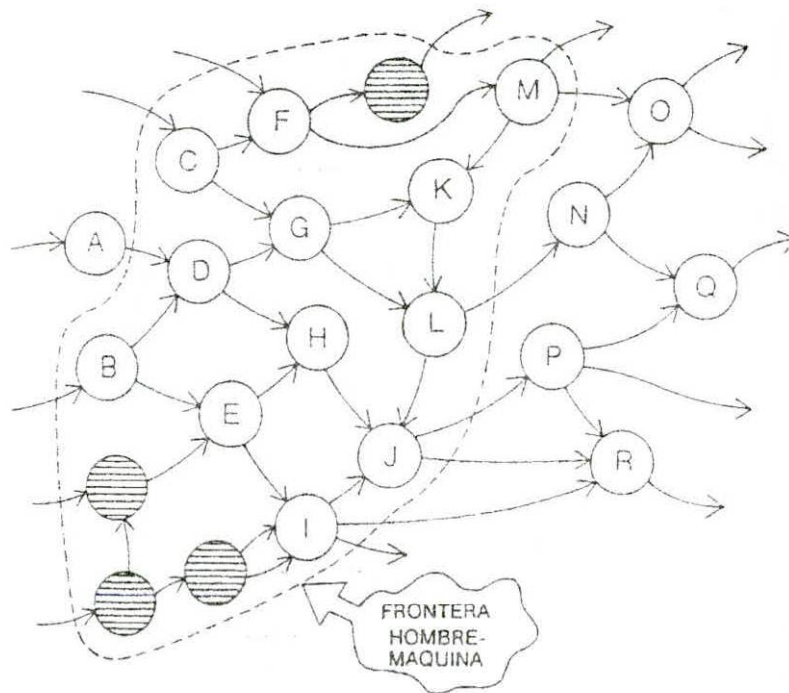


Figura 6.14: Fronteras H/M

6.5. Evaluar alternativas

La selección de una opción, de entre un conjunto de ellas, requiere de un estudio costo/beneficio para cada una de las alternativas.

Cada alternativa generada en la subfase anterior tiene asociados ciertos niveles de inversión, costos y beneficios que es preciso determinar a objeto de establecer cuál es aquella que resulta más ventajosa en función de los objetivos y restricciones, ligadas al proyecto informático que se está desarrollando.

Para establecer las cifras de costos y beneficios, existen algunas consideraciones básicas que es preciso destacar, entre ellas: costos de operación, de mantenimiento, de entrenamiento, de conversión de datos, de recursos humanos, de riesgo y los costos financieros entre otros. En relación a los beneficios, normalmente más difíciles de determinar, se consideran aspectos que van desde los ahorros asociados a la mayor eficiencia que se logra con la incorporación del procesamiento automatizado, como ocurre por ejemplo con la eliminación de facturas erróneas en una subfunción que procesa órdenes de compra para generar facturas, hasta el valor del prestigio asociado a la automatización.

En relación a las inversiones, es preciso considerar todos aquellos desembolsos en que se ha de incurrir, a objeto de obtener un sistema capaz de operar normalmente en el ambiente organizacional en el cual deberá desarrollar su quehacer.

Es decir, se incluyen desembolsos ligados a la adquisición de hardware y software y aquellos asociados a la construcción misma del sistema, tales como lo son los gastos en que se incurre durante las fases de desarrollo.

En otras palabras en esta fase se somete a evaluación económica cada diagrama de flujo de datos físico nuevo generado en la subfase anterior, obteniendo para cada una de las alternativas un indicador de rentabilidad que permita compararlas, en un proceso como el que describe la figura 6.15.



Figura 6.15: Evaluación de alternativas de automatización.

Los instrumentos de evaluación susceptibles de considerarse son preferentemente el valor actual neto (VAN), y/o la tasa interna de retorno (TIR).

6.6. Seleccionar alternativa

Una vez evaluada cada una de las opciones, es preciso seleccionar aquella que se estima ofrece las mejores garantías de eficiencia y eficacia.

Esta subfase se encuentra más allá del rango decisional del que se dispone durante la etapa de análisis, puesto que normalmente en la elección de la alternativa, existen otras consideraciones, fuera de los factores objetivos ligados a la evaluación económica que es preciso tomar en cuenta, como lo son los lineamientos de las políticas informáticas y organizacionales tal como lo muestra la figura 6.16



Figura 6.16: Selección de alternativas.

6.7. Configurar documentación final

Esta subfase consiste básicamente en ajustar la especificación de requerimientos -que en análisis estructurado se denomina justamente Especificación Estructurada- a las normas de documentación establecidas, dándole un sentido de integridad al trabajo y configurándolo como un documento final, lo que hace que esta fase tenga más bien un carácter orientado a la documentación.

Este ajuste consiste en conformar la especificación estructurada, es decir en la obtención de un documento de especificación coherente y explícito, de tal modo que permita enfrentar con efectividad la fase del diseño, la próxima etapa en el desarrollo de un sistema informático.

Este proceso de conformación puede, incluso, involucrar algunas tareas complementarias, muchas veces necesarias, tales como:

Incorporar los detalles que pudiesen haberse diferido.

Reconfigurar para destacar las interfases fundamentales.

Preparar una guía de la Especificación Estructurada para mejorar la efectividad de la especificación.

Estos aspectos se describen a continuación.

6.7.1. Incorporación de los detalles diferidos

Es conveniente que durante la fase de análisis se diferieran ciertos detalles relativos al nuevo sistema. La mayoría de ellos referidos a consideraciones técnicas, tales como formatos, esquemas y a la estructura interna del sistema, los cuales son aspectos que deben tenerse en consideración en las fases posteriores del desarrollo de un sistema. Sin embargo, existen otros aspectos que debiendo diferirse, se deben incluir en la Especificación Estructurada a instancia de los usuarios.

Se sugiere usar las mismas herramientas del análisis estructurado para incluir aquello que se obvió y que es preciso incorporar a la especificación.

Entre aquellos aspectos que fueron dejados de lado y que normalmente los involucrados en el quehacer de la función desean incorporar, se destacan:

- Mensajes de error: la existencia de flujos de error queda establecida al conformar los diagramas de flujo de datos, pero sin especificar sus características. Ahora se debe explicitar el texto exacto de cada mensaje.
- Inicio y término: los diagramas de flujo de datos no permiten reflejar la dinamicidad del sistema, ya que corresponden a una visión estática de él. Sin embargo, los responsables operacionales de la función podrían requerir que en ellos se expliciten las instancias de inicio y término. En el modelo esto se puede hacer incorporando un indicador al diagrama de flujo de datos con las inclusiones respectivas en el diccionario de datos. Obviamente que estas adiciones, en general, deben evitarse y su incorporación debe ser sólo excepcional.
- Información de control para satisfacer los requerimientos del usuario: el diagrama de flujo de datos configurado no contiene los flujos de control. Durante la fase de análisis se construye un modelo significativo que no explicita flujos de control, pero el usuario, en cambio, puede necesitar incorporarlos excepcionalmente, en determinados flujos claves para el desarrollo de sus actividades, deseando que se destaquen como tales en el diagrama de flujo de datos, y con notas que indiquen el significado de estos componentes adicionales.

- Incorporación de formatos: algunas veces el formato de un documento dado es tan importante para el usuario que éste no desea que finalice la fase de análisis sin que esté completamente especificado. Esto puede suceder con un documento de salida que, posteriormente, será usado como entrada para otro sistema, o con uno donde el formato está determinado por normas. En tales casos, es habitual que los usuarios insistan en que el formato actual se incluya en la Especificación Estructurada.
- Conversión: en el común de los casos normalmente el analista deja la conversión para las etapas posteriores del desarrollo del sistema. Sin embargo, hay importantes excepciones. Hay proyectos que requieren de la construcción de un completo sistema de conversión que permita llevar a cabo el proceso mismo de la conversión. Si ello es necesario, entonces dicho sistema debería especificarse en la fase de análisis.

Puede haber muchos aspectos susceptibles de incorporar en esta fase, pero es aconsejable no recargar la especificación con asuntos que son propios de la tarea del diseñador. Diseñar, por ejemplo, todos los formatos de entrada y salida, e imponerlos como parámetros para los desarrollos posteriores al análisis, constituye una sobreespecificación.

Por ejemplo, en relación a los formatos, en la fase de análisis lo importante es que el usuario verifique si está de acuerdo con el contenido y organización de los flujos de datos que se mueven entre los procesos, lo cual está documentado explícitamente en el diccionario de datos.

Al definir los formatos existen muchas otras consideraciones físicas que es preciso considerar, por lo tanto son los encargados de la tarea de implementación, quienes deben enfrentar el problema de elegir el formato más conveniente, de acuerdo a los requerimientos físicos que al respecto plantean los usuarios.

6.7.2. Presentación de las interfases fundamentales

Como parte de la tarea de configurar la Especificación Estructurada, es necesario reagrupar y regraficar el diagrama de flujo de datos de modo que todas las interfases fundamentales aparezcan en el primer nivel de desagregación de la jerarquía. En particular, es la interfase hombre-máquina la que debe quedar reflejada a este nivel. En términos concretos esto implica representar la porción automatizada del sistema como una única burbuja de primer nivel.

La figura 6.17 muestra el primer nivel del diagrama de flujo de datos de un sistema. En este caso las fronteras hombre-máquina corresponden a los flujos de datos que se mueven hacia o desde la burbuja 5.

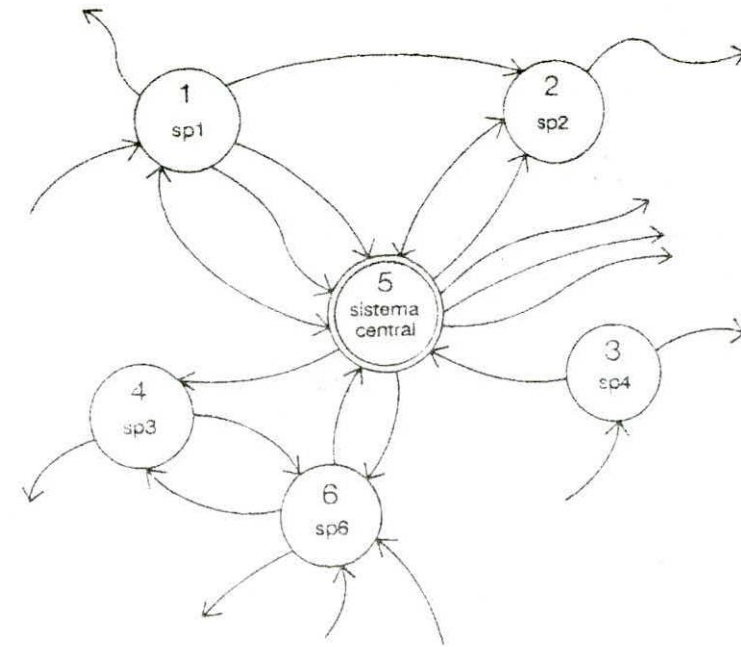


Figura 6.17: DFD para fronteras H/M a primer nivel

La burbuja 5, junto con todos los subprocesos que conforman la estructura jerárquica de aquella parte del diagrama de flujo de datos que referencia lo automatizado, más los flujos de datos asociados a cada una de las burbujas y que se incluyen en el diccionario de datos del sistema, y las especificaciones de proceso que rigen las transformaciones de las burbujas de más bajo nivel en la jerarquía, constituyen la especificación del sistema automatizado que será implementado.

Esta, junto al resto de las burbujas del primer nivel y sus respectivos niveles jerárquicos inferiores, conforman la documentación de la Especificación Estructurada.

Mostrar las fronteras hombre-máquina en el nivel más global del modelo, mejora indudablemente la lectura de la Especificación Estructurada. Hacerlo así permite destacar las relaciones entre lo automatizado y lo manual, e incluso es posible generar diagramas, similares a los diagramas de flujo de datos físico actual, que muestran los alcances del nuevo sistema tal como se aprecia en figura 6.18.

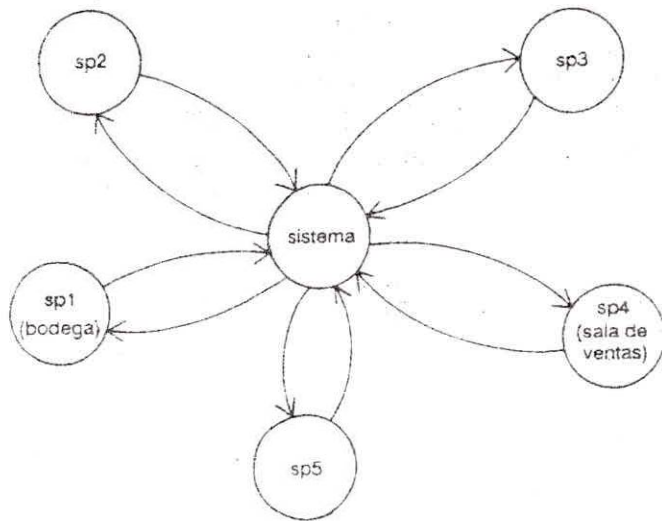


Figura 6.18: DFD para fronteras H/M a primer nivel

Los cuadros de organización, son de indudable utilidad para otras instancias funcionales en la Organización.

En el caso de un sistema distribuido también es posible mostrar las fronteras máquina-máquina en el primer nivel. La figura 6.19 muestra el primer nivel de desagregación de un sistema en una Organización que dispone de sucursales. Allí existe una instalación para una cierta sucursal que forma parte del sistema y otra en las oficinas centrales. El sistema requiere, para el cumplimiento de sus tareas, de un trabajo interactivo entre ambos nodos de la Organización. En este ejemplo, las fronteras hombre-máquina muestran el conjunto de flujos de datos de comunicación entre la porción manual del sistema (las otras burbujas) y la parte automatizada. Las fronteras máquina-máquina corresponden a los flujos de datos que pasan entre las burbujas 1 y 2.

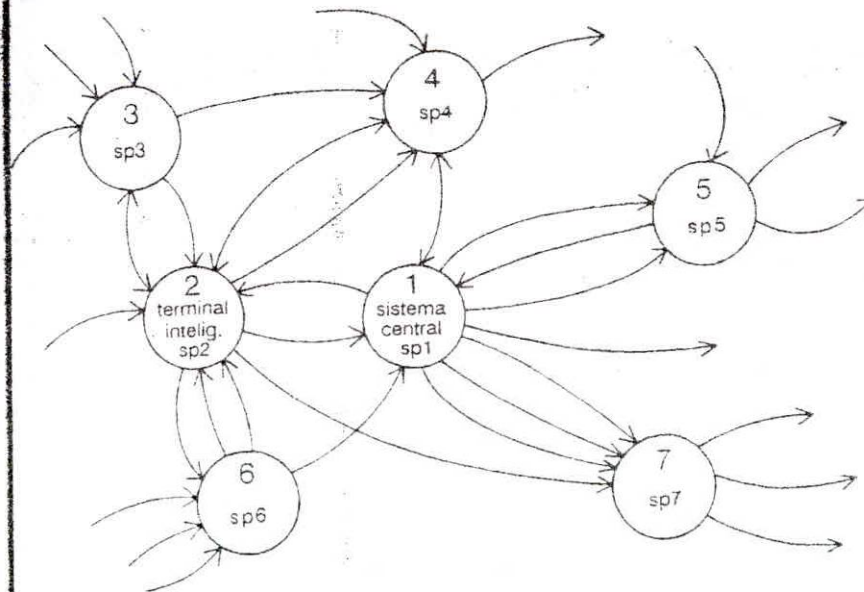


Figura 6.19: Fronteras H/M para un sistema distribuido

La metodología propuesta, por cierto que no postula su aplicación a modo de recetario, muchas veces existe la necesidad de incorporar nuevas notaciones o convenciones para ajustar las herramientas a las necesidades específicas de un determinado proyecto. Para ello debe enfrentarse con criterio profesional y evaluarse en términos de la efectividad requerida.

Por ejemplo, como señala Tom De Marco [DEM-79], a veces ocurre que ciertas personas insisten en configurar sus diagramas de flujo de datos con triángulos.

Cualquiera sea el grado de desviación de la forma de especificación que se ha expuesto, es preciso evitar fundamentalmente la redundancia y la sobre-especificación, ya que éstos entorpecen la eficacia y hacen disminuir la eficiencia de la Especificación. Podría pensarse, por ejemplo, en la conveniencia de tener un pequeño listado jerárquico que muestre todos los procesos y subprocesos que comprende el sistema. Pero obviamente, este documento adicional es redundante. En este caso cualquier cambio, además de considerarse en la documentación del sistema establecida por la metodología propuesta, debe también introducirse al documento adicional. Es decir, el esfuerzo de mantención se ve incrementado al dedicar tiempo y recursos a un componente que nada aporta a la efectividad de la especificación del sistema. Siempre que se incorpore algún nuevo elemento a la especificación, se está afectando el grado de libertad disponible para el diseñador y se corre el riesgo de incrementar el costo de la implementación. Tom De Marco [DEM-79] al respecto señala que: "Si el ámbito de acción de los diseñadores está determinado por la libertad de movimientos, establecida en función sólo de los requerimientos reales, podrán enfrentar de manera más favorable el control de los costos y, a la vez, maximizar la capacidad de mantención que ha de tener el sistema".

6.7.3. Una guía para la Especificación Estructurada

Un buen complemento a la documentación obtenida, es incluir un anexo no demasiado extenso que explique el contenido y los alcances de la Especificación Estructurada, describiendo sus herramientas y las convenciones que gobiernan su utilización.

Este anexo debería referirse entre otros, a los siguientes aspectos:

- Qué es la Especificación Estructurada.
- Los componentes de la Especificación Estructurada.
- Rol de los diagramas de flujo de datos.
- Convenciones para los diagramas jerárquicos de flujo de datos.
- Convenciones y notaciones en el diccionario de datos.
- Correlación entre los diagramas de flujo de datos y el diccionario de datos.
- Correlación entre los diagramas de flujo de datos y el conjunto de especificaciones de procesos.
- Interpretaciones de los diagramas de estructura de datos y diagramas de acceso inmediato.

6.8. La Especificación Estructurada

Uno de los objetivos del análisis estructurado es generar una especificación de la función organizacional sometida a análisis.

Esta especificación se denomina Especificación Estructurada y consiste en una descomposición exhaustiva de la estructura de la función sometida a desarrollo informático. Es decir, describe los componentes -subprocesos- y relaciones -flujos de datos- en términos de la conformación jerárquica que la determina como función.

Explicita además las subfunciones que se requiere desarrollen su tarea automatizadamente, es decir aquellas que se constituirán a la postre en objeto de software.

Se trata por lo tanto de, un documento cuyo contenido carece de consideraciones físicas, por cuanto privilegia el qué hacer más que el cómo hacer.

tiene básicamente dos intenciones. Una de ellas es describir el sistema para los involucrados en el quehacer habitual de la función y la otra es proporcionar a los diseñadores una visión amplia pero rigurosa y exacta del resultado final que deben alcanzar, es decir, explícita en este caso los requerimientos de modo que sea posible llevar a cabo a la postre, la construcción e implementación del producto de software que sustentará, en lo referente a procesamiento automatizado de datos e información, a la función o proceso objeto de tratamiento informático.

Una vez concluida la Especificación Estructurada, en otras palabras, una vez terminada la fase de análisis, se transfiere la responsabilidad de las tareas al grupo técnico que ha de enfrentar el diseño del sistema, tal como muestra la figura 6.20.

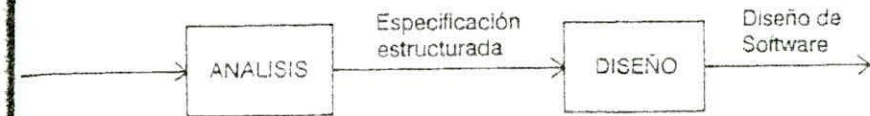


Figura 6.20: La Especificación Estructurada como interfase entre las fases de análisis y diseño.

La Especificación Estructurada forma parte de la documentación del trabajo realizado durante la fase de análisis, y como todo documento una de las características que debe presentar para ser efectivo es el no ser redundante. Como la redundancia es un fenómeno que se relaciona con la información involucrada en el documento, se hace necesario establecer primero el tipo de información que debe contener cada elemento que lo conforma, para ajustarse a aquella característica esperada.

De esta forma, y en un marco de mínima redundancia, la Especificación Estructurada contiene:

- La información relativa a las estructuras de los datos que maneja el sistema, expresada en términos de componentes e interrelaciones, que se incluye en diccionario de datos.
- La información relativa al procesamiento de los datos contenida en la descripción de los procedimientos que han de regir las tareas inherentes a los procesos.
- La información relativa a la relación entre los flujos de datos y los diferentes procesos que los afectan incluida en los diagramas jerárquicos de flujo de datos.
- La información relativa a los diferentes accesos inmediatos que se requieren

En consecuencia, la Especificación Estructurada es un componente fundamental de la documentación del sistema, y consta básicamente de cuatro elementos:

Los Diagramas Jerárquicos de Flujo de Datos: El Diagrama Jerárquico de Flujos de Datos muestra la descripción -por niveles de detalle- de la función o proceso explicitando las subfunciones y los flujos de datos que las relacionan, en un modelo que describe el sistema a diferentes niveles de agregación, desde lo más global a lo más detallado.

El Diccionario de Datos: El diccionario de datos es la descripción del contenido y la organización de cada uno de los flujos, y de cada uno de los archivos que aparecen en el diagrama de flujo de datos. En otras palabras, un Diccionario de Datos documenta la composición y la organización de los datos que conforman las interfases y los archivos.

La Descripción de Procesos: La descripción de procesos es la especificación de los procedimientos que rigen las transformaciones a que son sometidos los flujos de datos, al interior de cada uno de los procesos de más bajo nivel. En otras palabras, la Descripción de Procesos documenta los procedimientos que rigen las transformaciones de entradas en salidas en las burbujas del último nivel jerárquico del diagrama de flujo de datos.

El Diagrama de Acceso Inmediato: El Diagrama de Acceso inmediato es un modelo que refleja las diferentes opciones de acceso a los almacenamientos requeridos en la función o proceso sometida a análisis.

La figura 6.21 muestra un modelo en el cual se aprecian estos elementos, componentes de la Especificación Estructurada.

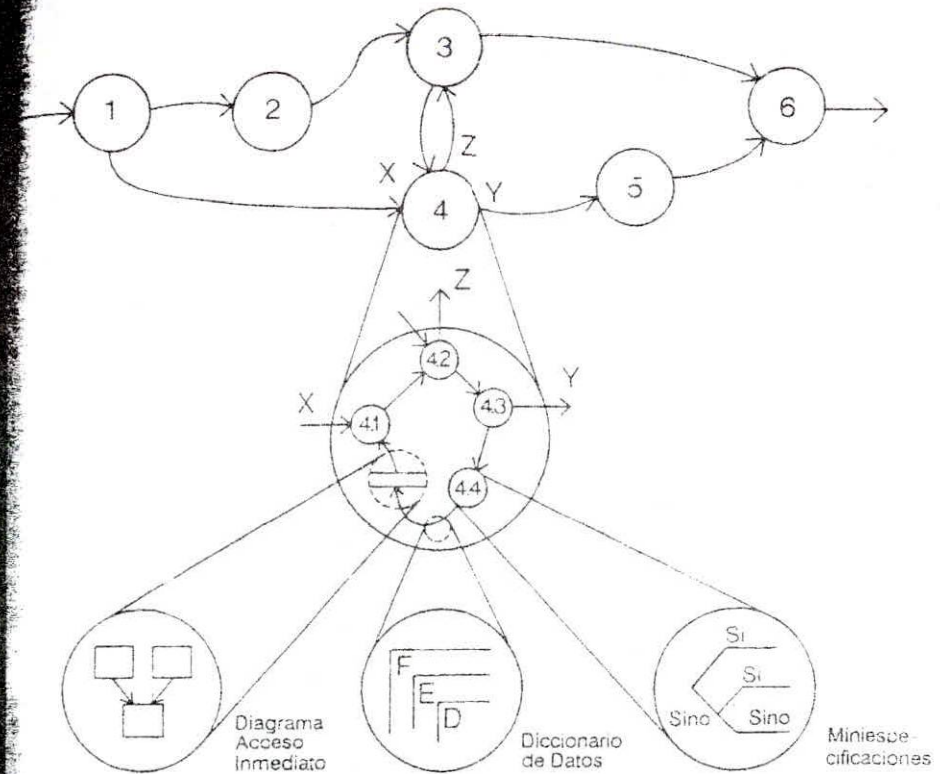


Figura 6.21: Componentes de la Especificación Estructurada

La Especificación Estructurada es un documento con ciertas particularidades, entre cuyas características cabe destacar:

- Es un instrumento de comunicación que facilita el intercambio entre el especialista y el personal que se desenvuelve en la función o proceso.
- Se trata de una representación gráfica de la función objeto de análisis. Un diagrama de flujo de datos presenta una visión de la función especificada mediante un conjunto de diagramas modulares jerárquicos, del tipo grafos, que se constituyen en una representación que permite una fácil comprensión del sistema, ya que lo describe tanto en perspectivas globales como detalladas.
- Es una representación particionada, lo cual permite el acceso a algunas subfunciones -partes del documento- sin perder la perspectiva del entorno más global en el cual se inserta.

El diagrama de flujo de datos es un modelo del sistema que lo representa en diferentes niveles de descomposición, desde la representación más global, en términos de entradas y salidas solamente, hasta las más detalladas que muestran las tareas operacionales, que son aquellas que efectivamente llevan a cabo las transformaciones entrada-salida.

El criterio de descomposición se basa en la noción de refinamiento sucesivo o particionamiento top-down, con el cual se conforma un modelo jerárquico recursivo del sistema, en donde los procesos o subfunciones son los elementos básicos en virtud de los cuales se muestra la desagregación del sistema. Ello implica, por lo tanto, un desarrollo progresivo de la explicación y descripción del sistema, que va desde lo más abstracto a lo más concreto.

- Es riguroso. Además del modelado de la función en los términos sugeridos por la descomposición top-down y que se refleja en los diagramas jerárquicos de flujo de datos, su diccionario de datos ofrece una documentación completa de las relaciones entre las subfunciones, especificando contenido y estructura de cada uno de los flujos y una completa descripción de los archivos, también en términos de contenido y organización. Su diagrama de acceso inmediato explicita las diferentes opciones de acceso a los archivos, y sus especificaciones de procedimientos, para cada uno de los subprocesos que llevan a cabo las tareas operacionales, corresponden a la descripción ortogonal y precisa de las reglas operacionales que rigen las transformaciones entrada-salida que cada uno de ellos debe realizar.
- Todos los elementos de la Especificación Estructurada -Diagramas de Flujo de Datos, Diccionario de Datos, Descripción de Procesos y Diagrama de Acceso Inmediato- están supeditados a ciertos estándares y convenciones para su especificación, y el hecho de ser en gran medida gráficos estructurados, les confiere a cada uno de ellos una exactitud y una facilidad de comprensión significativa, evitando con ello las ambigüedades y las interpretaciones equivocadas.
- Es mantenible. Su carácter modular, y los niveles mínimos de redundancia que manifiesta, facilita la incorporación de las modificaciones, que sea necesario introducir, en cualquier momento del ciclo de vida del sistema, lo cual le confiere niveles adecuados de eficiencia, al proceso de mantener actualizada la documentación dicho sistema.
- Es iterativo. Los componentes de la Especificación Estructurada se caracterizan, entre otros aspectos, por las facilidades que muestran para ser desarrolladas separadamente. Ello facilita las eventuales revisiones o redefiniciones que es preciso llevar a cabo, producto de nuevas consideraciones surgidas de las etapas posteriores del desarrollo del sistema.
- Es lógica, no física. La especificación estructurada no referencia a aquellos elementos del sistema que dependen del hardware, de los sistemas operativos o de las configuraciones en general.

Se remite específicamente a las funciones y subfunciones que tienen alcances orgánicos, por lo que responden más bien a la lógica del funcionamiento de la organización. Una de las razones por las cuales los alcances de esta especificación limitan con lo organizacional, es el evitar conformar descripciones de funciones o procesos vulnerables a los cambios que pueden producirse, justamente por las modificaciones que pueden sufrir la configuraciones de hardware y software básico.

Lo esencial de la Especificación Estructurada es el modelado del sistema. Sin embargo, ésta sólo muestra componentes y relaciones, ésta es la razón por la cual el diagrama de flujo de datos requiere del Diccionario de Datos, de la Especificación de Procesos y del Diagrama de Accesos Inmediatos como aspectos complementarios que no pueden obviarse.

Estos componentes de la especificación requieren, al igual que la construcción de los diagramas de flujo de datos, de ciertas pautas y criterios para su construcción, los cuales serán descritos exhaustivamente en los capítulos siguientes.

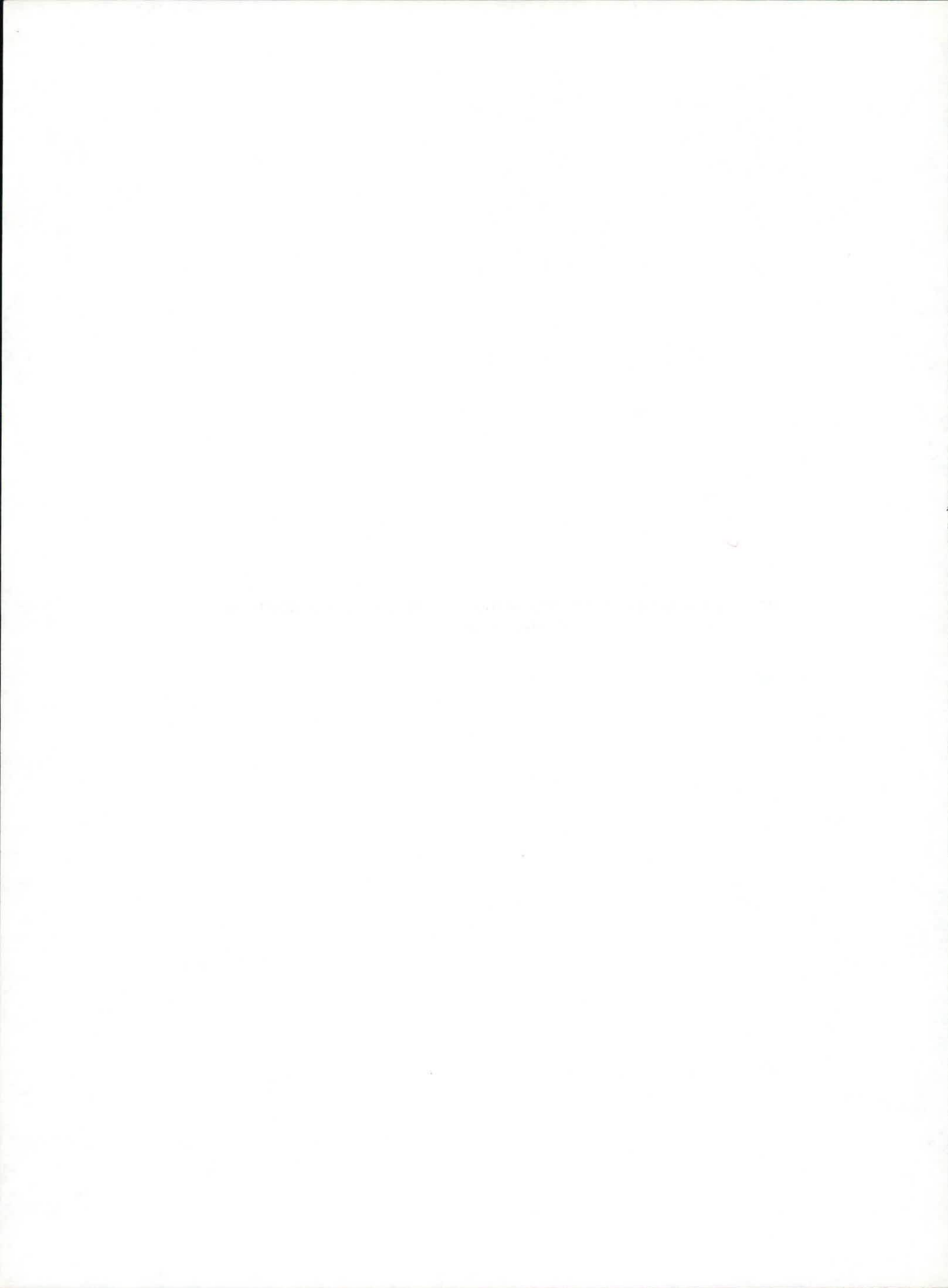
Referencias

- DEM-79, De Marco, T., Structured Analysis and Systems Specification, Yourdon Press, New York, 1979.
- MIL-56, Miller, G. A., The magical Number Seven, plus minus two: some limits on our capacities for processing information, Psychological Review, Vol. 63, Marzo 1956.

Otras Lecturas

- Freeman, P., Requirements Analysis and System Specification. Prentice-Hall, 1979.
- Gane, T., C. Sarson, Structured System Analysis, Mc Donnell Douglas, 1982.
- Gehani, N., D. Mc Gettrick, eds., Software Specification Techniques, Addison-Wesley, 1986.
- Pressman, R., Ingeniería de Software un enfoque práctico, Mc Graw-Hill, Madrid, 1988.

**IV. EVALUACION Y ADMINISTRACION PROCESO DE DESARROLLO
DE SOFTWARE**



RECOLECCION Y ANALISIS DE DATOS

Algunos tópicos de la parte anterior del texto muestran la necesidad que la organización de software se sostenga utilizando medidas y administración estadística, lo que significa la planificación cuantitativa y el control del proceso.

En esta sección se discute los siguientes dos temas:

- Los principios de la recolección de datos del proceso de software y las consideraciones claves para establecer un programa de recolección de datos. Estos datos satisfacen las necesidades básicas para el análisis cuantitativo y establece el fundamento para la planificación y administración de la calidad del software.
- El plan de calidad del software, porqué es importante y cómo es producido, analizando las medidas de la calidad, los planes de la calidad, estimaciones de la calidad y el control de la calidad.

I. PRINCIPIOS DE LA RECOLECCION DE DATOS.

Los principios de una efectiva recolección de datos son:

- El dato es recolectado de acuerdo con objetivos específicos y un plan.
- La elección de los datos a ser recolectados está basada en un modelo o hipótesis acerca del proceso que está siendo examinado.
- El proceso de recolección de datos debe considerar el impacto de la recolección de datos sobre la organización entera.
- El plan de recolección de datos debe tener un soporte administrativo.

1. Los objetivos de la recolección.

Debe haber un objetivo claro en la recolección de datos para mejorar el proceso. Es poco probable recolectar datos útiles sin ese objetivo y en forma accidental. *Hay 4 razones básicas para recolectar datos de software:*

- a) **Comprensión:** Los datos pueden ser recolectados para aprender acerca de algún ítem o proceso.
- b) **Evaluación:** Los datos pueden ser usados para estudiar algún producto (o actividad), para ver si satisface criterios de aceptación.
- c) **Control:** Los datos pueden ser usados para controlar alguna actividad.
- d) **Predicción:** Los datos pueden ser usados para desarrollar indicadores de estimación o tendencias.

2. Modelos de recolección de datos.

Para medir el proceso de software en forma exitosa, se debe comenzar por conocer los resultados esperados, para precisar acerca de los datos que son necesarios.

Después de algún estudio y evaluación, se puede hacer suposiciones acerca de los eventos claves del proceso y sus relaciones.

Los datos son luego recolectados para verificar estas suposiciones.

Las medidas están relacionadas a este modelo conceptual; las variaciones son examinadas y se hace cambios al proceso o al modelo para obtener un acuerdo.

Se puede tomar muchas medidas en un proceso de software complejo, pero sin un modelo del proceso la recolección de datos puede no entregar nada de valor.

Se puede encontrar relacionamientos entre eventos distintos, pero ello no tiene ningún valor predictivo. Sin una clara conexión causa y efecto, los estudios estadísticos no son útiles. Aún cuando se encuentren altas correlaciones, es necesario establecer una razón probable de qué está sucediendo antes que se pueda esbozar cualquier conclusión útil.

Es por esta razón que se dice que se requiere un modelo como una base para cualquier programa de recolección de datos útil.

3. El impacto de la recolección de datos en la organización.

Hay dos aspectos fundamentales que se deben considerar en los planes de recolección de datos: los efectos de las mediciones sobre la gente y los efectos de la gente sobre las mediciones.

En el primer aspecto, cuando la gente sabe que está siendo observada, generalmente su rendimiento cambia. Cuando se establecen medidas para la evaluación de la gente o cuando ellas sienten que están siendo evaluadas, se esfuerzan por tratar de superar las medidas, lo que resulta en un mejoramiento de los resultados esperados, ya que las motivaciones personales son muy difíciles de anticipar. Todas las mediciones del proceso de software deben ser no amenazantes y no evaluativas.

El segundo aspecto se relaciona con el impacto de la actitud de la gente sobre los datos. El trabajo actual de coleccionar datos es tedioso y generalmente debe ser hecho por profesionales de software que están ocupados en otras tareas. Si no existe un convencimiento de que los datos son importantes, ellos no recolectarán o no serán cuidadosos en hacerlo. También existe una tendencia natural a diferir el trabajo, obteniendo resultados inadecuados o incompletos.

Se debe mostrar a los profesionales cómo los datos pueden ayudarlos y se les debe motivar para realizar la recolección en forma oportuna.

Por razones de objetividad y costo, es importante automatizar la recolección de datos tanto como sea posible.

4. Soporte de la administración.

Dado que la recolección de datos es costosa y existe un retardo de los beneficios que entrega, ésta debe ser vista como una inversión y como tal es fundamental el soporte de la administración para suministrar los recursos necesarios.

Cuando los beneficios resultantes sean detectados por los administradores de proyectos, éstos estarán más dispuestos a soportar el trabajo.

II. EL PROCESO DE LA RECOLECCION DE DATOS.

Los analistas y programadores a menudo ven la recolección de datos como una costosa pérdida de tiempo. Los administradores de proyectos no ven el valor directo para sus proyectos y no están dispuestos a sacrificar sus prioridades inmediatas. Es por esto que el programa de recolección de datos debe comenzar con el soporte de la administración y debe ser cuidadosamente planificado y validado.

1. La administración del proceso de recolección de datos.

En un comienzo, la recolección de datos es demasiado costosa debido a la falta de experiencia. Con una mayor experiencia y mejores herramientas, los costos pueden ser disminuidos, pero nunca llegarán a ser insignificantes. Es por esto que el proceso de recolección de datos debe ser cuidadosamente planificado y administrado.

Los datos deben ser definidos de manera precisa para asegurar que se obtenga la información correcta y ésta debe ser validada para asegurar que representa el proceso en forma exacta.

Los datos deben ser almacenados y mantenidos por alguien que posea y administre la base de datos del proceso.

2. El plan de recolección de datos.

El plan de recolección de datos puede ser producido por el grupo del proceso de Ingeniería de Software, con la ayuda de los proyectistas y la participación de alguno de los profesionales que recolectarán los datos. El plan debe cubrir los siguientes tópicos:

a. Qué datos son necesarios, por quién y para qué propósitos.

El plan debe establecer clara y precisamente cómo serán usados los datos. Para prevenir cualquier insinuación de ser usados para evaluación personal, el plan debe enfatizar en el producto y para ayudar a los profesionales en mejorar el proceso, más que proveer información de control.

b. Cuáles son las especificaciones de datos.

Esto incluye la definición de los datos iniciales a ser recolectados y el plan para definir cualquier dato adicional que probablemente será necesario. Las definiciones deben ser revisadas y aceptadas por todas las partes involucradas, y todos los aspectos resueltos antes que comience la recolección de datos.

c. Quién recolectará los datos.

Las personas que recolectarán los datos deben ser identificadas; ellas aceptan hacer el trabajo y entienden qué hacer y cuándo hacerlo.

d. Quién dará soporte a la recolección de datos.

Se provee de un programa de entrenamiento y de un grupo de soporte que esté disponible para responder consultas.

e. Cómo serán recolectados los datos.

Debe disponerse de formularios apropiados y facilidades de entrada de datos y establecerse y documentarse procedimientos apropiados. También es conveniente hacer un ensayo de la recolección de datos antes de implementarla en forma apropiada.

f. Cómo serán validados los datos.

Ya que los datos del proceso de software son propensos al error y son volátiles, deben ser validados rápidamente. Se debe identificar y usar con cuidado los datos que no son validados.

g. Cómo se administrarán los datos.

Se puede recolectar grandes cantidades de datos en proyectos de software. Son necesarios un equipo de profesionales y un sistema apropiado de facilidades para asegurar la entrada oportuna y exacta, hacer modificaciones periódicas y dar soporte de recuperación.

3. Validación de datos.

Se ha detectado que los datos de software son altamente propensos al error, por lo cual es necesario contar con validaciones especiales y exhaustivas.

La manera en que los datos son validados depende de los datos involucrados. Con medidas de tiempo total transcurrido, puede ser comparado con el total de los elementos. Los volúmenes producidos (tales como líneas de código o defectos), pueden ser registrados en varios puntos del proceso y comparados. También se puede usar lugares de chequeo para clasificar en categorías de errores u otros ítems de juicios.

Cuando el personal ha sido entrenado sobre las definiciones o métodos de recolección de datos, es necesario establecer lugares de chequeo periódico. El personal que hace la validación debe ser capaz de explicar las definiciones de datos, los métodos de recolección de datos y los objetivos del programa.

III. MEDIDAS DE SOFTWARE.

1. Características de los datos.

Se estudia a continuación algunas características y clasificaciones de las medidas de software, como también algunas medidas específicas.

a. Características.

- Deben ser robustas: Esto es, deben ser repetibles, precisas y relativamente insensibles al menor cambio de herramientas, métodos o características en los productos. Cuando este no es el caso, es difícil decir si las variaciones son causadas por el proceso o por medidas anómalas.
- Deben sugerir una norma: Cuando se trata de medidas de defectos, mientras más se acerquen a cero, el valor es mejor.
- Deben referirse a productos específicos o propiedades del proceso: Esto es, deben referirse a errores, tamaños o recursos gastados.
- Deben sugerir una estrategia mejorada: Medidas de complejidad, por ejemplo, implican un objetivo de reducción de complejidad.
- Deben ser un resultado natural del proceso: Generalmente se requiere de mediciones especiales, pero se debe recordar que el intento del proceso es producir un producto, no hacer mediciones. Las medidas deben ser un producto para el normal desarrollo del trabajo, con solo un modesto esfuerzo adicional necesario para recolección.
- Las medidas deben ser simples: Las mediciones complejas son generalmente difíciles de explicar, aplicar e interpretar, y ellas a menudo indican un modelo del proceso extremadamente complicado.
- Deben ser predecibles y manejables: Las medidas son de mayor valor cuando son proyectadas en el tiempo y luego comparadas con las experiencias actuales. El personal del proyecto puede entonces ver mejor cómo cambiar su comportamiento para mejorar los resultados.

b. Clasificación.

Las medidas de software pueden ser clasificadas de varias maneras:

- Objetiva / Subjetiva: Distingue entre medidas que cuentan cosas de aquellas que involucran el juicio humano.
- Absoluta / Relativa: Las medidas absolutas generalmente son invariantes a la suma de nuevos ítemes. El tamaño de un programa, por ejemplo, es una medida absoluta y es independiente del tamaño de los otros. Las medidas relativas cambian, tal como un promedio o una pendiente de una curva.

Las medidas objetivas son a menudo absolutas, mientras que las subjetivas tienden a ser relativas.

- **Explícita / Derivada:** Las medidas explícitas son tomadas directamente, mientras que las derivadas son calculadas desde otras medidas explícitas o derivadas. Un ejemplo de una medida explícita es la cantidad de meses de programación gastados, mientras que una derivada es la productividad o líneas de código por mes de programación.
- **Dinámica / Estática:** Las medidas dinámicas tienen una dimensión tiempo, como errores encontrados por mes. Aquí, los valores tienden a cambiar, dependiendo de cuándo las medidas son hechas en el ciclo del producto. Las medidas estáticas, sin embargo, permanecen invariantes, como el costo total del esfuerzo gastado o defectos totales encontrados durante el desarrollo.
- **Predictiva / Explicativa:** Las medidas predictivas pueden ser obtenidas o generadas por adelantado, mientras que las explicativas son producidas después del hecho.

El objetivo de la Ingeniería de Software, en la recolección y análisis de datos, es usar un número creciente de medidas objetivas, absolutas, explícitas y dinámicas, para controlar y mejorar la manera en que se hace el trabajo. La exactitud de la predicción basada en esas medidas debe mejorar gradualmente, hasta que se asemejen estrechamente a la experiencia actual.

2. Medidas del tamaño del software.

Las medidas del tamaño son importantes porque la cantidad de esfuerzo requerido para hacer más tareas está relacionada directamente con el tamaño del programa involucrado. Generalmente, no existen medidas aceptadas del tamaño de programa que satisfagan todos los criterios descritos anteriormente.

Un problema es la ausencia de simplicidad: No existen medidas simples del tamaño del software porque el tamaño no es un sujeto simple. Por ejemplo, uno debe considerar nuevos cambios, eliminaciones, reuso y modificaciones de código. Existen también diferencias dependientes de si uno está tratando con lenguajes de alto nivel, código assembler, job control, código objeto, comentarios, definiciones de datos o presentación de pantallas.

Líneas de código. La medida en líneas de código (LOC) es probablemente la forma más práctica para mediciones de tamaño de programa. El mayor riesgo es que un uso descuidado de éstas puede motivar una maximización del contador de LOC. No existen medidas de tamaño de software que eviten esta maximización y la única solución es usar las medidas de tamaño como guías y no para evaluación de la gente u organizaciones.

Se ha descrito algunas maneras alternativas de contar líneas de código:

- Líneas ejecutables.
- Líneas ejecutables más definiciones de datos.
- Líneas ejecutables, definiciones de datos y comentarios.
- Líneas ejecutables, definiciones de datos, comentarios y JCL.
- Líneas físicas en una pantalla de entrada.
- Delimitadores lógicos, tales como punto y coma.

Aún con esas definiciones, se debe decidir qué tipo de líneas contar:

- Solamente nuevas líneas.
- Nuevas líneas y cambios.
- Nuevas líneas, cambios y reuso de líneas.
- Todas las líneas entregadas, más el código construido temporalmente.
- Todas las líneas entregadas, el bosquejo temporal y el soporte de código.

Medidas de LOC normalizadas. Otro problema con la cuenta del LOC es que con lenguajes de alto nivel no reconoce el contenido funcional incrementado de las líneas de código fuente. Algunas organizaciones compensan esto usando factores para generar lenguaje assembler equivalente. Para ser realmente representativos, tales factores deben reflejar tanto los lenguajes fuente y assembler que están siendo usados, como también las funciones del programa particular involucrado. A menudo, sin embargo, estos factores son tomados sin un fundamento adecuado. Esto cambia entonces una medida objetiva y explícita a una subjetiva y derivada. Por esto no es prudente usar tales factores en todo.

Una razón importante para usar medidas simples es hacer una máquina contable. Con el software comercial IBM System/370, las cuentas de LOC se hacen con un programa de medición común. Ya que los desarrolladores de IBM usan la misma herramienta de contabilidad, no hay duda acerca de la definición de la medición y cómo obtenerla.

3. Datos de errores.

Se ha notado que la determinación de lo que constituye un error depende en gran medida de lo que el programador entiende. Por ejemplo, en un experimento la contabilidad de errores fue hecha contando los cambios requeridos para corregir el programa. En un caso hay dos posibles correcciones, dependiendo de las intenciones originales del programador. Una alternativa tomó ocho cambios diferentes y el otro tomó solamente tres. Ambos programas resultantes fueron correctos. Se debe distinguir claramente entre errores hechos por los programadores y las correcciones defectuosas. Para hacer esto, se requiere de las siguientes definiciones:

Errores: Son equivocaciones humanas y su cuantificación a menudo depende del entendimiento de las intenciones del programador. En el caso de errores tipográficos y sintácticos, sus causas generalmente son claras, pero la naturaleza y causa de los errores de diseño son mucho más difíciles de establecer con precisión.

Defectos: Son condiciones impropias de los programas, que generalmente son el resultado de errores. No todos los errores producen defectos en los programas, como con comentarios incorrectos o algunos errores de documentación. Inversamente, un defecto puede resultar de causas que no son de programación, tales como empaquetamiento o manejo de programas.

Bugs: Un bug es un defecto de programa que es encontrado en la operación, ya sea bajo un test o en el uso. Los bugs resultan de defectos, pero no todos los defectos causan bugs (algunos son latentes, pero nunca encontrados). Con software ampliamente usado, los defectos pueden ser encontrados muchas veces, resultando en bugs duplicados o múltiples por defecto.

Fallas: Una falla es un mal funcionamiento de una instalación de usuario. Puede resultar de un bug, una instalación incorrecta, un golpe en la línea de comunicación, una falla de hardware u otros.

Problemas: Los problemas son dificultades encontradas por el usuario. Pueden resultar de fallas, mal uso o mal entendimiento. Los problemas son eventos humanos, como opuestos a las fallas, que son eventos del sistema.

El proceso de análisis del problema.

Básicamente, el problema de corregir errores de software es distinguir causas y efectos. Ya que se intenta tanto corregir como prevenir los errores que cometen los programadores, se debe seguir la cadena de posibles eventos medibles. Al testear una operación, los problemas encontrados deben ser analizados para ver si ellos resultan de fallas, sea que las fallas provienen de bugs de software y sea que estos bugs son causados por defectos en el programa. En la situación de mantención tradicional, estos defectos son reparados y el uso operacional es reasumido. Tanto en desarrollo, como en mantención, es deseable seguir los bugs para encontrar los defectos y errores específicos que los causaron. Encontrar errores de esta forma involucra una larga cadena de análisis, comenzando con el comportamiento operacional de un sistema completo. En algunos sistemas comerciales ampliamente usados, los defectos representan sólo cerca del 3% al 5% de los problemas de software encontrados por los usuarios. Este proceso, después del hecho de identificar los defectos del software y los errores que los causaron, puede ser bastante caro y consumidor de tiempo.

El proceso de análisis del problema demuestra una distinción fundamental entre testeos e inspecciones. Cuando se testea, los problemas son encontrados durante la operación y son analizados para determinar sus causas. Se puede así medir directamente el dato problema, pero se debe, a través del análisis, obtener o deducir los datos por el número de fallas, bugs, defectos y errores. Esto demuestra una ventaja interesante en el proceso de inspección del software: Los errores son identificados directamente. Esto ahorra tiempo; las medidas de error obtenidas de las inspecciones son inherentemente más válidas que los datos derivados del testeo operacional, debido a que se requiere poca opinión.

Clases de medidas de defectos.

Debido a que los defectos del software son de gran interés tanto en el testeo como en la operación, es natural usarlos como una medida clave del proceso. Los defectos del software (y los bugs que los identifican) pueden ser categorizados como sigue:

Severidad: Mide el impacto actual o anticipado de un defecto sobre el ambiente operacional del usuario. Generalmente tales medidas son valiosas para establecer prioridades de servicio.

Síntomas: Esta categoría se refiere al comportamiento observado del sistema cuando el defecto es encontrado.

Dónde encontrar: Los defectos pueden ser categorizados por la localización en el sistema donde fueron identificados. Esta es a menudo el segmento de programa que actualmente está siendo ejecutado en el momento en que el bug fue encontrado.

Cuándo encontrar: Categoriza los defectos por cuándo ellos son encontrados en el ciclo de vida del software, como en un test unitario, un test de función, un test del sistema, un test de aceptación, instalación u operación.

Dónde fue causado: A menudo el elemento del sistema que causa el problema no es aparente en el momento en que el bug es encontrado. Ejemplos de ello son las rutinas de chequeo, que identifican errores en las salidas de los otros programas. Mientras la rutina de chequeo estaba siendo ejecutada en el momento en que surgió el problema, ella no fue la causa del problema.

Cuándo fue causada: Esto generalmente requiere considerable más análisis. Generalmente identifica los defectos introducidos en el diseño de alto nivel, diseño detallado, implementación, corrección de defectos y otros.

Dónde fue fijado: Este es un registro de dónde fueron hechos los cambios cuando se fija la instalación.

Cuándo fue fijado: Mientras esta no es una medida de desarrollo primaria, puede ser útil para evaluar y mejorar la mantención y el proceso de crecimiento.

Cómo fue fijado: Está relacionado con cómo el cambio es diseñado y aplicado.

Todas estas son formas válidas para categorizar los bugs y los defectos. En Ingeniería de Software se está más interesado en obtener datos de dónde fue causado, cuándo fue causado y cómo fue casado tan rápida y exactamente como sea posible. Esto permite un producto más efectivo y mejorar el proceso.

Se ha publicado muchos sistemas de categorización, pero no hay ninguno que satisfaga todas las necesidades. Cada organización debe así desarrollar su propia clasificación, basada en su situación particular.

4. Datos de productividad.

Los datos de productividad son recolectados para responder alguna de las siguientes preguntas:

- ¿Cuánto esfuerzo se gasta en cada tarea del proceso de software? Estas clasificaciones de tareas son definidas en la arquitectura del proceso.
- Para cada elemento producido, ¿qué esfuerzo fue dedicado a cada tarea del proceso?
- ¿Cuál es el costo relativo de suprimir los defectos por los diversos tests e inspecciones y por cada categoría de error principal?
- ¿Cuán efectivos son los diversos métodos y tecnologías para mejorar la productividad de cada tarea del proceso?
- ¿Cuál es la historia del recurso para las principales tareas del proceso y cómo deben ser ajustados los factores planificados para proyectos subsecuentes?

Las medidas de productividad de Ingeniería de Software son difíciles de recolectar e igualmente engañosas de usar. Es ingenuo pensar que el rendimiento de los programadores puede ser medido por alguna métrica simple de productividad, tal como líneas de código por mes de programador. Cuando se aplica medidas de software simples, unidimensionales, el personal no está dispuesto a proveer ningún número por temor que sea mal usado.

Los datos de productividad pueden ser útiles en el nivel de tarea, pero llegan a ser progresivamente menos pertinentes cuando las tareas son combinadas en grandes trabajos unitarios. Cada equipo del proyecto coloca muchas condiciones únicas. Ya que no hay dos proyectos que sean comparables en un sentido grueso, ninguna medida simple puede ser útil para compararlos. Infortunadamente, cuando los administradores saben que un proyecto genera 1000 líneas de código mensual por programador y el otro solamente 200, ellos desean apurar al de baja productividad. El uso de un estándar común para dos proyectos diferentes es como comparar la construcción de un catillo de arena con la construcción del Castillo de Windsor,

contando las toneladas de arena usadas por hora de labor.

Productividad organizacional.

Las medidas de productividad están muy propensas al mal uso cuando son aplicadas a los niveles más altos del proyecto y de la organización. Es, sin embargo, deseable tener alguna medida de cómo es el rendimiento de la organización a través del tiempo. Para el software, infortunadamente, las únicas medidas simples disponibles son las líneas de código y el esfuerzo gastado. IBM ha usado el total de líneas de código anual enviadas por programador, como una medida de cada laboratorio de software. Esta medida tiene la ventaja de ser objetiva y fácilmente calculable, pero la desventaja es que no es muy robusta. Un pequeño cambio en el envío actual de un gran proyecto de diciembre a enero cambia dramáticamente el rendimiento percibido del laboratorio. Ya que esta es la única medida disponible, la administración decidió usarla para tener conocimiento de la tendencia histórica más que localizarla en las variaciones año a año o en comparaciones entre laboratorios.

Para usar la medida de productividad LOC, es esencial definir en forma precisa los términos usados. Para líneas de código, una herramienta contadora fue provista para el uso en todos los laboratorios. La cuenta de trabajo incluyó a todos los profesionales de software y al personal de soporte, incluyendo la administración. La mantención y el control de calidad no fueron incluidos. A pesar de que la medida ha tenido muchas fallas, no se ha propuesto una mejor.

El valor de tal medida gruesa de productividad es discutible. Mientras la tendencia provee un discernimiento útil; es difícil explicar todas las calificaciones importantes a los ejecutivos que no son muy versados en software. Se ha encontrado que los datos de productividad más útiles son los obtenidos como base para la planificación del proyecto. Ya que estos datos son relativamente finos en detalle, representa más cercanamente el rendimiento actual de las actividades. También provee una validación esencial de los parámetros de estimación usados para la planificación del proyecto.

Sensibilidad de las medidas de productividad.

Después de ejecutar un breve estudio para determinar porqué los números de productividad de software varían tan ampliamente entre las diversas compañías, se ha detectado diferencias significativas en las definiciones de línea de código y trabajo gastado en las organizaciones. Las diferencias entre las definiciones de línea de código pueden explicarse fácilmente por variaciones de al menos dos órdenes de magnitud en la productividad:

- La diferencia entre contar solamente líneas nuevas y cambiadas versus contar todas las líneas se calcula por un factor sobre 10. Para el producto liberado siguiente,

algunas organizaciones cuentan todas las líneas de programa enviadas, aún pensando que solamente un pequeño porcentaje de ellas involucra trabajo de desarrollo significativo.

- Cuando el código ha sido reusado muchas veces, algunas organizaciones cuentan cada reuso, algunas cuentan solamente la primera versión reusada y otras sólo cuentan los cambios requeridos por cada adaptación. Estas consideraciones se aplican bien a módulos macro, rutinas de librería y otros. Las variaciones observadas tienen un rango tan alto como diez o más veces.
- Las variaciones del lenguaje a menudo son compensadas por un cálculo del código assembler equivalente. Como se notó previamente, los factores arbitrarios son usados generalmente tal que puedan ser tan altos como seis o más, infortunadamente con pequeña justificación.

Para el trabajo gastado, las variaciones no son muy importantes, aunque pueden ser grandes.

Definiendo el trabajo tal que incluya solamente programadores de desarrollo directo como oposición a todas las categorías de trabajo involucradas, puede causar variaciones de productividad de tres a cinco o más veces.

Es claro que los datos de productividad pueden ser mal manejados, a no ser que todos los parámetros sean definidos y controlados estrictamente. Si esto no se hace y si los administradores usan tales medidas para propósitos de evaluación, los números a menudo pueden ser ajustados para verse bien. Los programadores tienen el mejor equipo para la generación y manipulación de datos, tal que de ellos puede esperarse que produzcan todo lo que la administración desea ver.

IV. ANALISIS DE DATOS.

Mientras muchos tipos de análisis son posibles, son pocos los que indican algunas formas en que los datos pueden ser usados para soportar el desarrollo y la mantención del software. Cuando las organizaciones recolectan datos y comienzan a usarlos, encontrarán muchas formas de aplicarlos.

1. Análisis de datos de error.

Se hizo un estudio exhaustivo del análisis de datos en el sistema operativo IBM-DOS. Este estudio identificó los tipos de errores cometidos durante el desarrollo de DOS y provee antecedentes para detectarlos y prevenirlos.

El proyecto particular estudiado fue una modificación del módulo 522 de DOS. Resultó en 190.000 instrucciones y 60.000 líneas de comentarios, de los cuales 86.000 instrucciones fueron cambiadas o agregadas.

Durante el estudio, un total de 740 problemas fueron encontrados por el test, con 432 clasificaciones como errores

de planificación únicamente. El balance fueron errores de operador, errores de máquina, sugerencias de mejoramiento, dificultades de documentación o duplicidades. Los efectos de estos errores en los 422 módulos cambiados permitió notar que los tres módulos más grandes tenían los números de errores más grandes. Cada uno de ellos tenía sobre 3000 instrucciones, mientras que el tamaño promedio del módulo era de 360 instrucciones. En total, sólo el 40% de los módulos tenía de todos los tipos de error, mientras que el 21% de los módulos que tenía más de un error contado entre el 78% del total de errores.

Con respecto a la densidad de errores de los módulos de DOS nuevos y modificados, se concluyó que el código cambiado tenía un 60% de razón de error más alta que el código nuevo (7.8 versus 4.8).

Se concluyó de estos datos que solamente cerca de la mitad de los errores podían ser evitados con mejores herramientas y técnicas de programación. Para el resto, el mejor medio era dado por la definición del problema, mejor entrenamiento y mejor disponibilidad de soluciones previas de problemas conocidos. Se encontró que el beneficio importante de recolectar y analizar datos es el foco provisto en las áreas con las mayores probabilidades de causa de problemas.

2. Análisis de datos de prueba.

Se puede hacer muchos análisis diferentes a los datos de prueba. Generalmente se está mirando el dato de error más que examinar las causas del error por defectos de programación. Algunos ejemplos de los tipos de preguntas que puede hacerse son:

- ¿Cuál es la relación entre la cobertura del test por el módulo del programa y el número de defectos detectados y que permanecen en el programa?
- ¿Cuál es la efectividad de cada tipo de test para encontrar los principales tipos de defectos y cómo esta efectividad cambia con el tiempo?
- ¿Cuál es el costo de encontrar cada tipo de defecto por cada categoría del test?
- ¿Cuál es la historia del defecto de un producto dado y dónde se debe localizar la atención para mejoras más efectivas?

Con un poco de experiencia, muchas otra áreas provechosas de estudio pueden ser sugeridas. Ya que los datos de prueba pueden ser obtenidos en forma relativamente fácil, el testeo es uno de los mejores lugares para comenzar a recolectar y analizar el proceso de Ingeniería de Software.

3. Análisis de las razones de inspección.

Se ha publicado un interesante análisis de datos de inspección. Describe el uso de la estadística para rastrear, estimar, planificar y catalogar el desarrollo y el trabajo QA en programas FORTRAN. Da normas acerca de las razones de inspección de diseño y en el código. Las desviaciones significativas de estos valores fueron la causa por la cual se revisó la forma en que la inspección fue conducida.

Es costumbre contar el tiempo de preparación como la suma de los tiempos de preparación de todos los individuales involucrados. Inversamente, el tiempo de inspección es el tiempo gastado por el equipo completo en el proceso de inspección. Para un grupo de inspección de cinco personas, una hora de inspección es igual a cinco horas de inspección de programador, mientras que una hora de preparación es igual a una hora de preparación de programador. Aunque no es totalmente claro, de la publicación se puede deducir que se usó un promedio de los tiempos de preparación y no la suma.

Como resultado de este estudio, se obtuvo algunas relaciones interesantes, entre las que se destacan las siguientes:

- Los errores encontrados por KLOC decrecen con el incremento de las razones de inspección. Mientras que es verdadero que código con un gran número de errores es difícil de revisar, esto se puede hacer en efecto en 400 LOC por hora. Aparece, de esta forma, que una aproximación sobre el límite de 300 a 400 LOC-hora está probablemente cerca del óptimo para estos programas en FORTRAN.
- El tiempo de preparación de la inspección debe ser aproximadamente igual al tiempo de revisión.
- Se hizo un estudio de los límites superior e inferior de las razones de inspección con varios tiempos de preparación, concluyendo que para el código, en una razón de preparación de 7 horas por persona por KLOC, los rangos de la razón de inspección van desde un bajo de 100 a un alto de cerca de 300 líneas por hora. Esto se iguala a una razón de 3 a 10 horas de inspección, más 7 horas de tiempo de preparación, o un rango total de 10 a 17 horas por KLOC. Para esta razón de preparación, el LOC por hora de inspección más preparación está así entre 60 y 100.

4. Análisis de datos de inspección.

Se ha provisto algunos datos sobre inspección de diseño y codificación, que pueden ser analizados con técnicas estadísticas. La siguiente tabla muestra algunos datos sobre esto:

SELECTED DESIGN INSPECTION DATA

Component	LOC	LOC	LOC	ERRORS	ERRORS	PH
	PH	IH	TH	TH	KLOC	IH
1	161	171	83	1.1	12.7	1.5
2	144	347	102	.5	4.7	2.4
3	108	50	34	1.1	31.3	2.2
4	764	788	388	2.0	5.2	1.0
5	102	149	60	.4	6.5	1.5
6	191	257	110	2.1	19.4	1.3
7	547	523	267	1.2	4.5	1.0
Average	288	326	149	1.2	4.7	1.5
SD	241	236	120	1.0	9.3	.5
UCL	770	798	389	3.2	30.7	2.6
LCL	0	0	0	0	0	.4
Total product(8)	178	245	103	1.4	13.2	1.4
Worst modules(9)	62	230	49	2.0	40.6	3.7

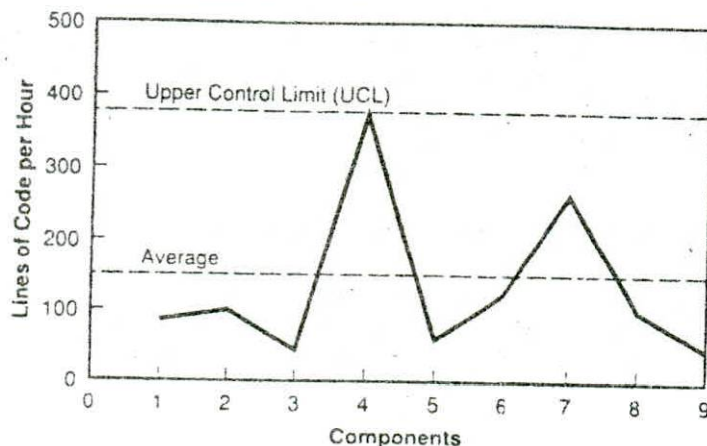
LOC: lines of source code
 PH: preparation hours
 IH: inspection hours
 TH: total inspection and preparation hours
 SD: standard deviation: $[\text{average of (values)}^2 - (\text{average of values})^2]^{1/2}$
 UCL: upper control limit = average + (2 x SD)
 LCL: lower control limit = average - (2 x SD) (0 if negative)

Estos datos fueron tomados de un gran proyecto con siete componentes principales. Cada componente tiene muchos módulos, algunos de los cuales son inspeccionados. Este dato es el promedio de las inspecciones por módulo por cada componente. Mientras es preferible usar el dato nivel del módulo directamente, este dato promedio sirve para demostrar las técnicas estadísticas involucradas.

Como se describió en capítulos anteriores, varios datos de razón y error pueden ser recolectados en cada inspección. El dato de la tabla anterior es: el total de líneas de código inspeccionado por programador-hora, los errores encontrados por programador-hora y los errores encontrados por KLOC. En adición, el dato promedio fue provisto para el proyecto total (componente 8) y para aquellos módulos considerados bastante pobres se reinspeccionará (componente 9). Este dato compuesto, sin embargo, no está incluido en los cálculos estadísticos siguientes. Los ítemes adicionales al final de esta tabla son explicados en los siguientes párrafos.

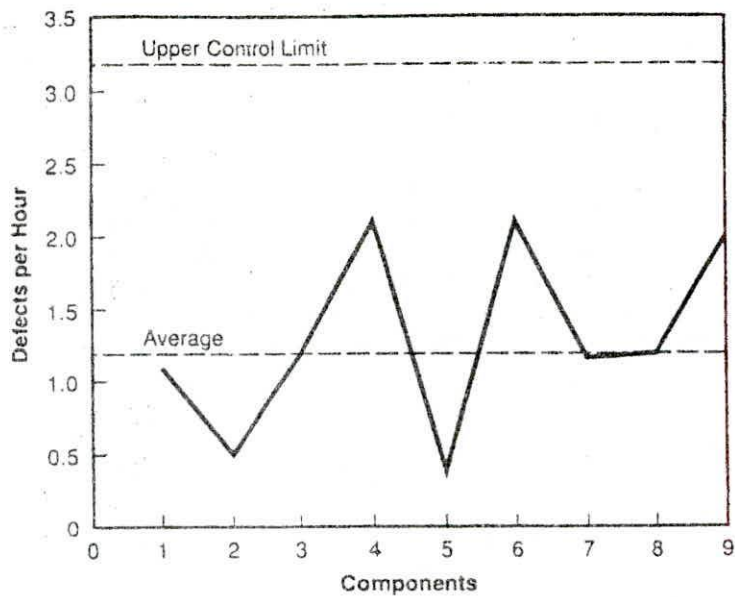
Construcción de diagramas de control estadístico.

Para facilitar el análisis, se construye los siguientes diagramas de control estadístico:



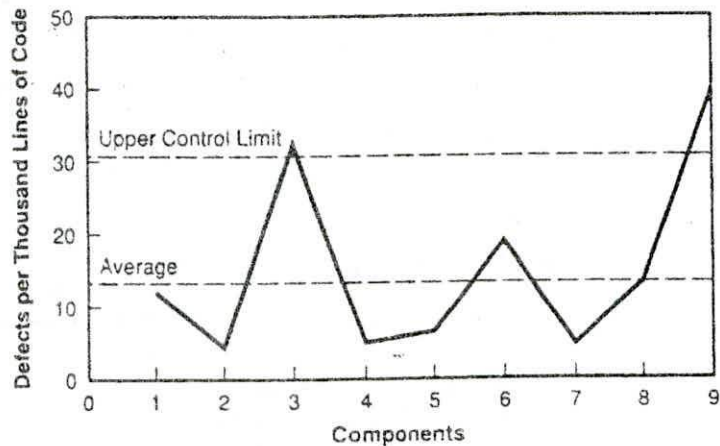
Inspection Control Chart—Lines of Code Inspected per Hour

Figura 1



Inspection Control Chart—Defects per Total Inspection Hours

Figura 2



Inspection Control Chart—Defects per Thousand Lines of Code

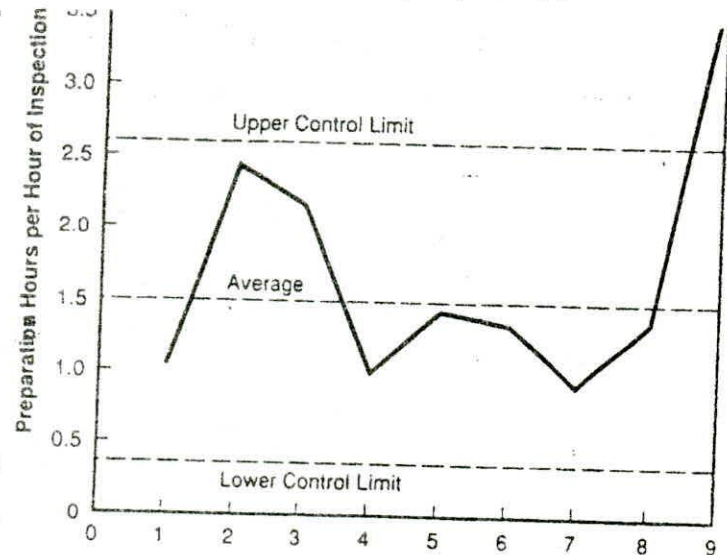
Figura 3

Estos gráficos se construyen usando los datos de la tabla anterior, junto con el promedio y los límites de control (UCL y LCL). Para encontrar estos límites de control, se hace los cálculos mostrados en la tabla: El promedio es el promedio matemático simple de los valores para los componentes 1 hasta 7 y el UCL y LCL son establecidos en dos desviaciones estándar antes y después del promedio. En instancias en las cuales estos cálculos pueden dar un valor negativo, se usa cero.

Estos límites de control son establecidos como una guía para ayudar a determinar qué resultados probablemente son debidos a la variación estadística random y que representan probablemente un comportamiento no usual. Por ejemplo, si se estuviera muestreando el precio preciso de objetos de una libra, habría pequeñas variaciones estadísticas alrededor del valor nominal de una libra. Si los cálculos fuesen hechos del valor promedio y la desviación estándar, los valores de peso actual estarían distribuidos en forma random alrededor del promedio, con cerca del 68% cayendo dentro de dos desviaciones estándar.

Uso de diagramas de control.

La razón de porqué los valores UCL y LCL son importantes es que se necesita identificar las condiciones no usuales y controlarlas cuando se trata de mejorar el proceso. Así, es importante resguardarse de aquellos valores que probablemente son debidos a razones puramente random y examinar el resto. Si, por ejemplo, se establece el UCL y el LCL en una desviación estándar anterior y posterior al promedio y luego se examina cada caso que caiga fuera de esos límites, se puede hacer una cantidad de trabajo no productivo. En promedio, aún con un proceso bien controlado, el 32% de los valores puede caer fuera de estos límites, debido a causas random. Los datos de las figuras anteriores y la siguiente proveen un ejemplo de cómo puede hacerse dicho análisis.



Inspection Control Chart—Hours of Preparation per Hour of Inspection

Figura 4

La figura 4 ha sido agregada debido a que provee antecedentes adicionales en el proceso de inspección. Sus datos también son derivados de la tabla dada. Primero, las variaciones mostradas en estas figuras son inusualmente grandes, indicando un proceso que está a lo más bajo tenue control estadístico. Las razones de inspección total tienen una variación de 10 a 1 y los errores por KLOC varían por sobre 7 a 1. Claramente, la primera etapa debe ser extender las razones para estas oscilaciones extremas y luego dirigir las. Este problema generalmente es resoluble con un programa de entrenamiento de inspección y enfatizar a la administración para ejecutar buenas inspecciones.

Aún con dichas oscilaciones naturales, se puede esbozar algunas conclusiones acerca de las inspecciones individuales. Las inspecciones para la componente 1 aparecen cerca del promedio en cada respecto. Para la componente 2, los errores encontrados por hora y por KLOC son relativamente bajos, mientras el tiempo total de inspección fue casi normal. La figura 4 provee antecedentes adicionales: Una gran parte del tiempo de preparación fue requerido por hora de inspección. Esto puede ser interpretado de varias formas: (a) una inspección muy bien preparada y llevada a cabo, (b) una inspección normal de código complejo, o (c) un grupo inexperimentado tratando de entender el programa. En las situaciones b o c se puede aplicar algún examen adicional.

La componente 3 es una situación completamente diferente. Aquí, el tiempo de preparación fue grande, la razón de inspección fue baja y se encontró un gran número de errores. Esto se ve como una inspección cuidadosa de un producto de baja calidad. La componente 4 también dice algo diferente: una alta razón de inspección, bajo tiempo de preparación y un alto número de defectos encontrados por hora, pero bajo número de defectos encontrados en el producto. Esto se ve como una inspección pobremente preparada y ejecutada.

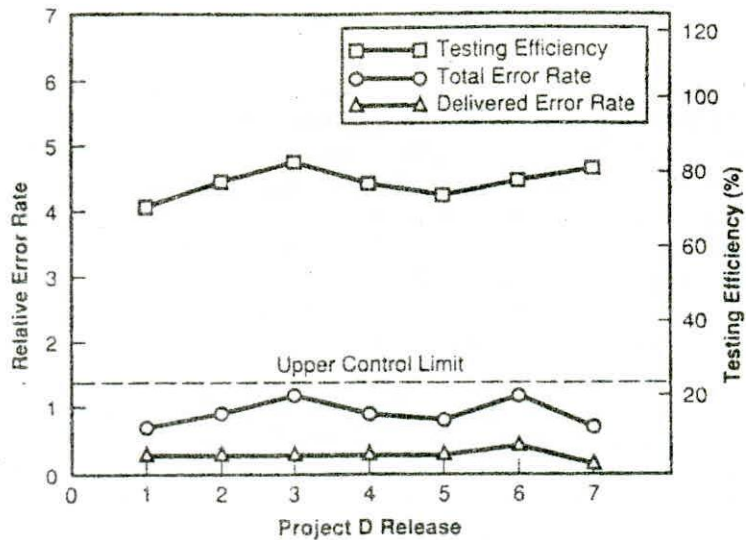
Se puede hacer un análisis similar a cada componente. Es interesante notar que los módulos identificados por el proyecto como difíciles tienen diagramas de control no usuales (rotulado como la componente 9 en las figuras). Ellas tienen las razones de inspección más bajas y la razones de preparación más altas. En el balance se ve como si estas inspecciones fueron bien hechas y que encontraron código de baja calidad.

Los datos de este orden pueden ser muy valiosos, particularmente cuando se acumulan sobre un período de tiempo en un gran número de proyectos. Con un poco de experiencia, se puede reconocer inspecciones que deben ser repetidas y producen elementos que necesitan estrecha atención. Analizando posteriormente estos datos contra el rendimiento producido durante el test y campo de uso, es posible establecer estándares informados por aquellas relaciones clave que indican el potencial producido o problemas del proceso. Si los límites de control exceden estos estándares, entonces el proceso de inspección mismo necesita atención. Inversamente, si los límites de control están dentro de estándares aceptables, el foco debe estar en las inspecciones individuales que caen fuera de estos límites. Finalmente, los datos de inspección y test deben ser examinados continuamente para ver si se necesita ajustar los estándares. Debido a las grandes variaciones entre las organizaciones y entre los productos de software, tales datos deben ser recolectados por cada organización para su propio uso.

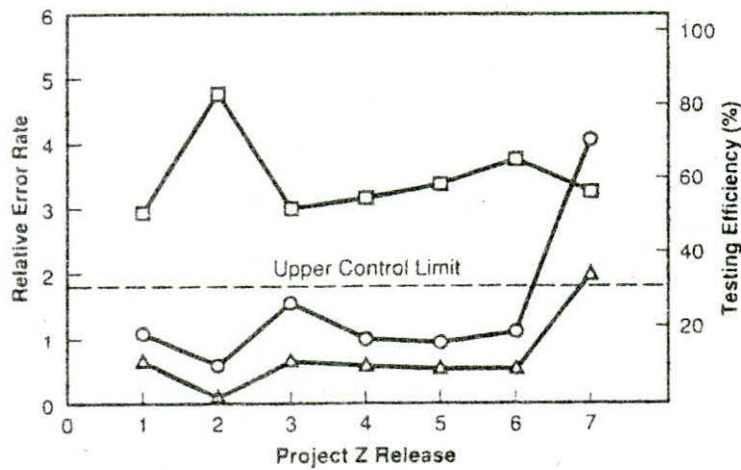
También es útil recolectar estos datos y usarlos para monitorear el rendimiento del proyecto. Esto provee una indicación de la dirección del rendimiento del proyecto y una indicación oportuna de problemas en el proceso. Una tendencia segura en uno o más parámetros, por ejemplo, puede proveer tempranamente una indicación de un problema. Aún si todos los valores estuvieran dentro de los límites de control establecidos, una tendencia puede indicar un comportamiento no random que puede ser direccionado.

5. Análisis estadístico de los datos de test.

Se da un ejemplo de cómo las técnicas estadísticas anteriores pueden ser aplicadas a los datos de test.



A. Project in Statistical Control



B. Project Out of Statistical Control

Process Control Results [5]

Figura 5

La figura muestra las razones de test para siete distribuciones consecutivas de dos productos, cada uno de los cuales tiene varios cientos de miles de líneas de código en tamaño. El proyecto D fue hecho bajo control estadístico, mientras el proyecto Z parece haber quedado fuera de control en la séptima distribución.

En las figuras, la línea tope (eficiencia del test) es el porcentaje de todos los defectos encontrados durante el testeo del sistema. En efecto, mide la calidad de la actividad de testeo. La próxima curva, razón de defecto total, muestra todos los defectos no arreglados reportados durante el testeo del sistema., testeo de aceptación y operación. Al evaluar la séptima distribución, los límites de control fueron calculados de las primeras seis distribuciones.

En este ejemplo, los datos para calcular estos diagramas no estuvieron disponibles hasta después del test de aceptación y un periodo del campo de operación. Se puede hacer diagramas similares de datos disponibles durante el desarrollo y test. Esto puede permitir hacer juicios sobre lo adecuado de cada módulo, cada inspección y cada test. Con tal dato, aquellos módulos que aparecen fuera de control pueden ser reinspeccionados o retesteados. Cualquier inspección o etapa del test que queda fuera también puede ser examinada para ver qué lo causó a desviarse y hacer las correcciones o mejoras indicadas al proceso.

V. OTRAS CONSIDERACIONES.

Ya que es fácil obtener conclusiones erróneas desde medidas pobremente ejecutadas, es importante recordar algunos puntos claves:

- Ya que generalmente los datos pueden ser interpretados de muchas formas, atender a lo que los datos dicen, más que usarlos para reforzar opiniones preconcebidas. Si el dato se usa para soportar sus vistas previas, sea suspicaz. Tener un modelo es esencial, pero creer que esto está implícito puede ser un error.
- Proveer recursos para validar y analizar los datos. La validación a veces es mirada en menos y el costo casi siempre es subestimado.
- El simple acto de recolectar datos cambiará el proceso. Después de un periodo razonable, los programadores se acostumbrarán a recolectar datos y los resultados se estabilizarán, pero en el medio tiempo recuerde que el proceso está cambiando mientras está siendo medido.
- Comience cuidadosamente y no intente hacer mucho de una sola vez. Cuando sea posible, integre la recolección de datos a las tareas existentes del proceso y automaticelo tan pronto sea práctico. Coleccionar datos es trabajo y los programadores se resistirán si se intenta hacer mucho muy rápido.
- Considere las opiniones de los recolectores de datos en el proceso y considere cualquier modificación, valorización o soporte que ellos necesiten. No prometa más de lo que puede entregar.
- Entregue los resultados iniciales a la gente relacionada con los datos antes de darlos a cualquier otro.
- Comenzar con un plan *completo*. Este debe incluir definiciones de datos, formularios y protocolos para registrar el dato, un sistema para coleccionar la información, un sistema de validación y planes para analizar y distribuir los resultados.

- Los debates sobre las definiciones y formatos de los datos pueden no estar terminados. Hasta un punto dado, éstos son útiles, pero una vez que todos los aspectos han sido tocados, poco más puede ganarse del estudio y debate. La única forma de encontrar qué datos deben ser recolectados y analizados es recolectar y analizar datos. Con esta experiencia, los debates pueden ser establecidos rápidamente.
- Obtener los recursos y obtener la partida.

VI. RESUMEN.

El dato del proceso de software es recolectado para aprender cómo mejorar el proceso. Los principios de una recolección de datos exitosa son: El dato es recolectado con un objetivo específico, la elección del dato está basada en un modelo del proceso que está siendo examinado, el proceso de recolección de datos mismo es definido y administrado, es diseñado conforme a las necesidades de la organización y debe tener el soporte de la administración.

El plan de recolección de datos especifica quién usará los datos y cómo serán usados. Cubre porqué el dato es necesario, las especificaciones del dato, quién y cómo lo recolectará y cómo será validado y administrado.

Para que sean de más valor, las medidas del proceso de software deben ser robustas, sugerir una norma, relacionarse al producto específico o a las propiedades del proceso, sugerir una estrategia de mejora, ser un resultado natural del proceso y ser capaz de proyectar y conducir.

Las medidas también pueden ser objetivas o subjetivas, dinámicas o estáticas y predictivas o explicativas.

El objetivo de la Ingeniería de Software es usar un número creciente de medidas objetivas, absolutas, explícitas y dinámicas, para controlar y mejorar la forma de hacer el trabajo. Como lo demuestran varios ejemplos de medidas de software, los datos pueden ser recolectados y analizados de varias formas y el conocimiento resultante puede ser aplicado para mejorar los productos y el proceso.

Algunas cosas claves a considerar en la recolección y análisis de datos son: Atender a lo que los datos dicen, proveer los recursos adecuados y no usar los datos para evaluar a la gente. Ya que el debate sobre las definiciones y formatos de los datos puede no tener término, la única forma de encontrar cómo recolectar y analizar datos de software es recolectar y analizar datos de software. Obtener recursos y la partida.

ADMINISTRANDO LA CALIDAD DEL SOFTWARE

Una de las mejores formas de evaluar una organización de software es examinar la calidad de sus productos. La calidad del producto es así una medida clave del proceso de software. Provee un claro registro del progreso del desarrollo, una base para establecer objetivos, y un esquema de trabajo para la acción actual. Cuando nos esforzamos por mejorar la calidad de nuestros productos de software, es instructivo examinar las experiencias en otras áreas. Mientras los métodos no son directamente relevantes al software, Agresti ha apuntado que los principios de la Ingeniería Industrial pueden ser aplicados efectivamente al software. El concluye que de todas las lecciones de la Ingeniería Industrial, la más importante involucra medidas de calidad y el impacto de tales medidas sobre el rendimiento de la gente que hace el trabajo.

En cualquier proceso intensivamente humano, la motivación es la clave para un buen trabajo. Para lograr un rendimiento consistentemente superior, los administradores deben establecer objetivos desafiantes de calidad y esforzarse para satisfacerlos. En caso contrario, si un administrados antiguo tolera un trabajo pobre, la mediocridad penetrará en la organización entera. Esto es verdad no exactamente para productos complejos de software, sino para todos los aspectos de un negocio. Cuando las entregas son siempre tarde, los reportes de estado son inadecuados, los memos de los administradores tienen errores tipográficos, los programadores asumen que la calidad no es una prioridad. Deba ser entonces difícil de sorprender si alguno de ellos no son cuidadosos con las inspecciones o test. Ellos saben que en un administrador exigente, probablemente, se dejará fuera la forma rápida y fácil. Con software, sin embargo, las soluciones rápidas y fáciles raramente compensan a largo plazo. La pregunta clave para el administrador es: ¿Usted intenta hacer algo acerca de la calidad del software?. Si lo hace, entonces su organización debe vivir por cuatro principios básicos de calidad:

1. A no ser que usted establezca objetivos de calidad agresivos, nada cambiará.
2. Si estos objetivos no son numéricos, el programa de calidad permanecerá tal como se habla (únicamente).

3. Sin planes de calidad, solamente usted esta confiando en la calidad.
4. Los planes de calidad son apenas papel, a no ser que usted los conduzca y los revise.

1. El Paradigma de la Administración de la Calidad.

Los principios básicos en la administración de la calidad del software son tales como aquellos para la administración de costos: Usted establece objetivos, hace planes, guía el rendimiento, y entonces ajusta el plan. Este ciclo es el fundamento para la ingeniería industrial, y se aplica igualmente bien al software.

2. Ejemplos de Calidad.

Los programas de calidad realmente funcionan. Se hace mención a una experiencia con un programa de monitoreo de un proceso en tiempo real, de 27.719 líneas de código fuente. Siguiendo rigurosamente un plan de calidad, las razones de densidad (promedio) de fallas fueron reducidas de 1.3 a 0.072 defectos por 1000 líneas, o un 95% de mejoramiento. Sus razones de productividad típicas, tenían un promedio de cerca de 7 líneas de código fuente por día de staff en proyectos previos, pero siguiendo el plan mencionado, ellos produjeron 29, una enorme mejora en la productividad.

Mientras ellos atribuían estos resultados a una combinación de muchas acciones administrativas y técnicas, las mejoras de calidad y productividad resultaron de encontrar y fijar esencialmente todos los errores tempranamente en el programa. Solamente, un defecto (bug) fue encontrado en el test del sistema, y que había sido forzado por conducir el sistema más allá de su razón de capacidad. En varios meses de uso los ocho clientes encontraron sólo 11 problemas. Los autores concluyeron que esto resultó de un programa de calidad medido y planificado.

3. Motivación de la Calidad.

El propósito de un programa de calidad es motivar la acción, para asegurar que esta acción resulte en mejores productos, las medidas deben representar razonablemente la bondad en términos del cliente.

Cuando es considerada como un programa motivacional, las medidas de calidad se toman con un significado especial. Por ejemplo, muchas métricas diferentes pueden ser usadas para caracterizar sistemas de software. En un reporte no menos de 22 parámetros son descritos para medir algunas de las 25 características deseables de un programa. Mientras algunas de estas pueden ser usadas como estándares de diseño, pocas pueden tener algún valor motivacional concebible. Esto no significa que no sean útiles. Una razón para tales medidas fue mostrada por un estudio que correlacionó un total de 15 características de software con la visión de varios profesionales experimentados de software. Ocho de las características tenían alta correlación con sus percepciones de la calidad del programa.

Las características que alcanzaron los más altos valores, fueron: independencia de dispositivos, autocontenidos, seguridad, robustez y consistencia.

Aún cuando es difícil ver cuántas de tales características pueden ser incluidas en un programa de medición de la calidad, su valor potencial no debe ser ignorado. Incluyendo estas características en los estándares del producto y diciendo a los profesionales que se desea y por qué, hay una buena opción que el trabajo correcto será efectuado.

3.1. Medición de la Calidad.

Un enfoque diferente es necesario para motivar un esfuerzo de calidad agresivo. Ya que la gente puede responder solamente un poco a la conducción motivacional de una vez, la medición y conducción para un programa de calidad debe estar altamente localizado. La necesidad es establecer un pequeño número de medidas de calidad numéricas específicas. Ejemplos típicos, son los defectos encontrados en varias etapas del proceso, la eficiencia de la inspección, o la cobertura del test.

Si no se usan medidas de calidad numéricas, solamente una medida de rendimiento del desarrollo permanece: el inventario. Desafortunadamente, cuando la adherencia al inventario es el único signo visible de progreso y no hay indicación de la eficiencia (suficiencia) del producto que está siendo producido, todas las energías son dirigidas a satisfacer el límite de tiempo. Esto ha motivado históricamente acciones que tienden a reducir la calidad del producto. Una focalización considerando sólo el

inventario, desvía la atención de los resultados y, sin algunas medidas de calidad, cualquier cosa enviada puede contar (ser registrada). Ya que cualquier medida sensible de inventario debe tener calificaciones de calidad, estas calificaciones son el fundamento para el programa de calidad.

Ya que ninguna medida sola puede caracterizar un producto complejo, un enfoque en un elemento del producto probablemente maximizará este indicador solo a costa del resto. Por otro lado, muchas medidas pueden aún llevar a una confusión. El problema puede ser resuelto con unas pocas medidas cuidadosamente seleccionadas como indicadores de calidad, mientras se insiste que todos los estándares y especificaciones aplicables sean satisfechas. Ya que el uso simplista de criterios numéricos puede causar esfuerzos no naturales para maximizar lo que se mide, la atención estrecha del administrador y el juicio informado deben ser siempre la guía final.

3.2. Clases de Medidas de Calidad

Las medidas de calidad caen en las siguientes clases generales:

- Desarrollo: Defectos encontrados, actividad de cambios.
- Producto : Defectos encontrados, estructura del software, estructura de la información (documentación), tests controlados.
- Aceptación: Problemas, esfuerzo para instalar, esfuerzo para usar.
- Uso : Problemas, disponibilidad, esfuerzo para instalar, esfuerzo para usar, opiniones del usuario.
- Reparación: Defectos, recursos gastados.

Como muestra la Tabla 16.1 estas medidas son caracterizadas como sigue:

- Objetivas: ¿Puede la medida ser producida repetidamente por gente diferente?
- Oportunas: ¿Están disponibles en tiempo para afectar el proceso de desarrollo o mantención?
- Disponibles: ¿Cuán difícil es obtenerla?.

- Representativas: ¿ En qué grado representa la visión del usuario de las bondades del producto?.
- Controlables: ¿En qué extensión pueden cambiarse su valor por acciones de desarrollo o mantención?.

TABLE 16.1
CLASSES OF QUALITY MEASURES

	OBJECTIVE	TIMELY	AVAILABLE	REPRESENTATIVE	CONTROLLABLE
Development					
Defects	yes	yes	yes	moderate	yes
Change activity	yes	yes	yes	poor	no
Product					
Error Seeding	moderate	yes	difficult	doubtful	moderate
Software structure	depends	yes	moderate	doubtful	yes
Controlled tests	moderate	yes	difficult	good	yes
Acceptance					
Problems	no	late	yes	good	moderate
Install effort	moderate	late	difficult	good	yes
Usage					
Problems	no	late	yes	good	moderate
Operating effort	moderate	late	difficult	good	yes
Surveys	no	late	difficult	very good	no
Availability	yes	late	moderate	very good	yes
Repair					
Defects	yes	late	yes	moderate	yes
Repair effort	moderate	late	moderate	moderate	yes

Claramente, es deseable que las medidas sean fácilmente disponibles, fáciles de recolectar, representativas de las necesidades del usuario, y controlables directamente por desarrollo o mantención. Desafortunadamente, no se conoce ninguna medida que sola satisfaga todos estos criterios. Por ejemplo, datos defectuosos están fácil y tempranamente disponibles en el proyecto, razonablemente objetivos y relativamente no caros de recolectar, pero no siempre representan las percepciones de calidad del cliente, inversamente, aquellas medidas que mejor representan al cliente a menudo, son caras de recolectar, o son subjetivas, o no están disponibles hasta que el programa está bien avanzado.

4. Criterio de Medición.

Decidir que medidas usar es esencial para considerar los objetivos del programa de medición. Si las medidas serán usadas para administrar el desarrollo de software, estas deben ser objetivas, oportunas, disponibles y controlables. Mientras deben tener alguna relación razonable con las necesidades del cliente, si ellas no pueden ser obtenidas durante el desarrollo, ellas no serán usadas.

Por otra parte, si las medidas de calidad son para soportar decisiones sobre la aceptación del producto, ellas deben representar razonablemente las necesidades del usuario y estar disponibles antes de la decisión en la aceptación. Mientras las medidas de defectos pueden tener algún valor, los mejores resultados generalmente pueden ser obtenidos con testeos controlados. Si las medidas de calidad usadas durante el desarrollo están disponibles, pueden ayudar en las decisiones de aceptación, pero al criterio principal debe estar basado en medidas más representativas.

4.1. Medidas de Defectos.

Los contadores de defectos solo representan moderadamente la visión del cliente de la calidad del producto. Mientras sea generalmente cierto que los programas con gran número de defectos ocultos (bugs) tienen pobre satisfacción del cliente, una vez que el nivel base de la calidad es alcanzado, las medidas de defectos no predicen mucho de la satisfacción del cliente.

Esto fue demostrado por un estudio hecho por un suministrador de software de seis programas. Estos programas fueron clasificados subjetivamente por un panel de usuarios de acuerdo a sus opiniones de la calidad completa del programa. Estos datos, junto con los datos relativos sobre errores de programa reportados por el usuario, y la actividad completa al problema del usuario, es dada en la Tabla 16.2

TABLE 16.2
 USER QUALITY RANKINGS

		RANKINGS	
	PANEL	DEFECT	PROBLEM
	OPINION	DENSITY	ACTIVITY
PRODUCT			
1	1	6	3
2	2	5	1
3	3	1	5
4	4.5	2	2
5	4.5	4	6
6	6	3	4
CORRELATION		-0.55	0.38

Esta tabla muestra que el programa que clasificaba mejor de acuerdo a las opiniones subjetivas de los miembros del panel, clasificaba muy mal en defectos por 1000 líneas de código. Las clasificaciones del problema, por otro lado, fueron algo más cercanas a estas visiones subjetivas. Las correlaciones relativas de todas las columnas muestran que las opiniones de los miembros del panel actualmente tienen una pequeña correlación negativa con las razones de defectos del programa.

Estas correlaciones negativas y débiles, con el problema y datos de defectos parecen indicar que, más allá de cierto punto, el problema y la actividad de defectos es menos importante que otros factores, tal como la usabilidad, al rendimiento y la funcionalidad. Claramente, determinar que representa verdaderamente la calidad a los ojos del cliente no es una tarea simple.

4.2. Actividad de Cambios

La actividad de cambios puede ser una medida útil de la calidad del desarrollo. Cuando la actividad de cambios permanece alta hasta tarde (casi al finalizar) en un programa de desarrollo, es una buena indicación del total de los problemas de calidad. Desafortunadamente, esta medida no es directamente controlable por lo general por el grupo de desarrollo, pero es una secuencia de factores tales como razones de defectos o estabilidad de requerimientos. Es así generalmente prudente usar medidas de defecto más directas para la planificación de la calidad, y conducir la actividad de cambio por módulos de programas como un indicador agregado del progreso en el desarrollo.

Los datos de la actividad de cambios pueden ser útiles si apropiadamente registrados. Las razones para cada cambio deben ser anotadas, junto con su tamaño. Por ejemplo, para cambios debido a correcciones de defectos, el contador de cambio del LOC a menudo es una medida más objetiva que el contador de defectos mismos. La razón es que; el alcance de un sólo defecto, a menudo es una decisión subjetiva. Si,

por ejemplo, un cambio de 8 líneas puede ser contado como un defecto en un caso y dos en otro, la medida no es repetible. Otro problema se relaciona con el manejo de los defectos que se extienden a varios módulos. ¿Es cada módulo acreditado con un sólo defecto, una fracción de un defecto, o es el defecto solamente asignado a uno de los módulos? Contar los cambios LOC resuelve esta confusión

Un beneficio similar puede resultar de contar los cambios de LOC debido a los cambios de requerimientos. Realizando esta actividad a través de todo el proyecto, el impacto del costo de la inestabilidad de los requerimientos puede ser demostrado gráficamente.

4.3. Errores Latentes

El error "latente" es una forma potencialmente interesante para evaluar la calidad del programa. La idea es inyectar un número conocido de defectos "mudos" (ó fantasmas) en el programa y entonces determinar cuantos de ellos pueden ser encontrados por varias pruebas o inspecciones. Si, por ejemplo, el 60% de los defectos latentes son encontrados, la presunción es que el 60% d los otros defectos (bugs) pueden ser encontrados también.

Mientras haya una evidencia creciente de que la idea latente es promisoria, debe verse aún como una técnica experimental. El problema clave es la creación de defectos que representen apropiadamente los defectos naturales que permanecen en el programa. Mientras varios enfoques han sido tratados, el "latente" permanece más como un arte que como una ciencia.

4.4. Estructura del Software

Varias medidas de complejidad han sido propuestas para evaluar diseños de software. La evidencia disponible sugiere, sin embargo, que la complejidad no es una mejor guía para niveles deseables de defectos, como es LOC.

4.5. Pruebas controladas

Con pruebas (test) controlados el producto está sujeto a un ambiente de trabajo simulado, y se hace una evaluación de su conformidad operacional. Las pruebas de factores humanos, por ejemplo, son usados típicamente para determinar lo adecuado de las interfases de usuario y la documentación; un ambiente especial de testeo se establece con la documentación, y las facilidades requeridas del sistema para demostrar aspectos específicos del comportamiento operacional. Esto difiere de la prueba del sistema en que algunas funciones del sistema pueden no ser necesarias o pueden ser suministradas a través de la simulación o prototipos.

4.6. Medidas del Problema

Los datos de actividad del problema proveen una medida potencialmente útil de la calidad del programa. Después de la instalación del programa y de la puesta en marcha, se pueden guardar registros de los problemas generalmente no son todos relacionados con los defectos del software, aquellos que lo son, son pasados a la organización de la mantención para su corrección. El resto generalmente se relaciona con los errores de los usuarios, problemas del hardware, defectos duplicados, mejoras sugeridas, y causas desconocidas. Para grandes sistemas con muchos usuarios, el número de problemas debido a los defectos del software es generalmente menor que el 3 al 5 por ciento de este total.

Las medidas del problema, sin embargo, tienen moderada correlación con la satisfacción del producto por el usuario. Esto es bastante entendible debido a problemas de desperdicio del tiempo y del dinero por parte del usuario, y generalmente causan muchos inconvenientes. Los contadores de problemas son una medida potencialmente atractiva de la calidad del software debido a que ellos incluyen defectos del software como también aspectos de la instalación y operacionales. Las dos desventajas más grandes con los contadores de problemas, son que: ellos solamente están disponibles después que el desarrollo está listo, y que son subjetivos. Su subjetividad se relaciona con la naturaleza humana de los problemas, los que pueden ser causadas por falta de comprensión humana, un simple error humano o por un mal funcionamiento del sistema. Cuando son conducidas apropiadamente, las pruebas controladas pueden a veces conducir estas desventajas y proveer información valiosa

antes de que el trabajo de desarrollo este completo. Pero tales pruebas generalmente son posibles en las etapas (tardías) del ciclo de desarrollo.

4.7. Esfuerzo de Instalación y Operacional

Mientras el esfuerzo requerido para instalar y operar el sistema es un buen indicador de la satisfacción del cliente, tales medidas de nuevo no están disponibles hasta la completación y la aceptación del cliente. Ellas también requieren que el cliente haga un montón de trabajo. En este caso, sin embargo, los factores que hacen cara la instalación y la operación, pueden ser examinados en forma útil tempranamente en el desarrollo del programa. El enfoque general es definir los procedimientos etapa por etapa, para instalar y operar el sistema y entonces medir actualmente el tiempo y esfuerzo requerido para ejecutarlas. Cuando estos test son conducidos por profesionales con factores humanos calificados, se puede aprender mucho acerca de las necesidades de entrenamiento, documentación adecuada, problemas de interfaz, y aún aspectos relacionados con el plan físico. con este conocimiento, los planes pueden ser revisados con los usuarios con conocimiento, y tests fuertemente controlados pueden ser conducidos. Mientras ha habido poca experiencia con tales planes de calidad, unos pocos juicios sugieren que es un área prometedora para mayor atención.

4.8. Examinar la Satisfacción del Cliente

Para entender la visión de los productos por parte del usuario, algunas organizaciones usan exámenes de la satisfacción del cliente. IBM, por ejemplo, envía cuestionarios periódicamente por correo o por teléfono para determinar las visiones de sus clientes de ciertos productos y servicios de soporte. Ellos incluyen en las preguntas, una petición para sugerencias o donde los productos o servicios pueden ser mejorados. Tales cuestionarios, si son apropiadamente conducidos, pueden proveer una visión informada de la calidad del producto. Desafortunadamente, esta medida nuevamente no está disponible hasta después de completar el desarrollo.

4.9. Confiabilidad y Disponibilidad

Para algunos programas, el cliente esta más relacionado con la habilidad del sistema, para ejecutar la función intentada cuando la necesita. Esto es llamado "disponibilidad". Tales medidas son particularmente importantes para sistemas y programas de comunicación que son fundamentales para la operación del sistema completo. Estas generalmente incluyen el programa de control, el administrador de base de datos, el plan de trabajo, la interfaz de usuario, control de comunicaciones, el administrador de la red, y el sistema de entrada-salida. La clave para incluir cualquier programa en esta lista, es que si este falla, echara abajo las aplicaciones críticas. Si es así la disponibilidad puede ser considerada.

La disponibilidad no puede ser medida directamente, pero debe ser calculada desde tales medidas probabilísticas como un tiempo medio entre las fallas (MTBF) y el tiempo medio requerido para reparar y restaurar el sistema para completa operación (MTTR). Suponiendo que el sistema es requerido para estar continuamente disponible, la disponibilidad es el porciento del tiempo total que el sistema está disponible para uso:

$$\text{Disponibilidad} = \left(1 - \left[\frac{\text{MTTR}}{\text{MTTR} - \text{MTBF}} \right] \right) \times 100$$

Claramente, si el sistema puede ser reparado instantáneamente entonces, cercando la disrupción operacional, la disponibilidad puede ser del 100%, considerando poco cómo ha fallado. Es así importante usar esta medida en conjunción con otras que se relacionan con los niveles de defectos o razones de fallas. La disponibilidad es una medida útil de la calidad de algunos sistemas. Desafortunadamente, es muy difícil proyectar previamente el testeo operacional. Ya que muchos productos, factores operacionales, y facilidades de servicio están generalmente involucrados, es también difícil asignar objetivos de disponibilidad a los elementos del sistema.

4.10. Seleccionando Medidas de Calidad

Mientras las medidas de calidad más apropiadas dependen de un uso entendido, ellas también dependen de los tipos de datos disponibles. Ya que el dato de defecto es tal, que la mayoría de las organizaciones de desarrollo de software pueden obtener antes que el sistema parta, debe ser usado. Mientras esta propiedad enfatiza la necesidad de reducir el número de defectos liberados, no debe considerar muchos otros atributos de calidad importantes.

Cuando ninguna medida cuantitativa de calidad este disponible para cosas tales como: funcionalidad, rendimiento, y factores humanos, otras etapas se deben tomar para asegurar que a estos tópicos importante se ha dado una atención adecuada. Algunas etapas prácticas son: incluir clientes representativos seleccionados en revisiones de especificación, conducir pruebas de utilización del laboratorio, hacer pruebas de benchmark de programas de clientes y conducir cuestionarios a los clientes. En edición, casi siempre vale la pena hacer alguna prueba temprano en otro ambiente operacional simulado o real, previo a la entrega final. Aún con los planes más pensados y un equipo de desarrollo altamente capaz, el ambiente operacional siempre parece presentar algunos problemas no esperados. Estos tests también proveen un medio para validar tempranamente los planes y pruebas de instalación y operacionales, haran mediciones disponibles tempranamente, y chequee la instalación, operación y procedimientos de soporte.

5. Estableciendo un Programa de Calidad del Software.

Los principios de calidad del software, establecidos al comienzo de este Capítulo, pueden ser detallados en los siguientes pasos:

- 1.- Un administrador experimentado (senior) establece objetivos agresivos de calidad, numéricos y explícitos. Sin medidas numéricas, el esfuerzo por la calidad será otro programa motivacional más con un pequeño impacto al final.
- 2.- Las medidas de calidad usadas son objetivas, requiriendo un mínimo de juicio humano.

3. Esas medidas son definidas y documentadas en forma precisa, de tal forma que se puedan hacer programas computacionales para recolectarlas y luego procesarlas.
4. Un plan de calidad es producido al comienzo de cada proyecto. Este Plan se somete a objetivos numéricos específicos, y esto es modificado en cada cambio significativo del proyecto.
5. Esos planes son revisados por complacencia con los objetivos de calidad del administrador. Cuando no se encuentra la complacencia, la replanificación o la aprobación excepcional es requerida.
6. La calidad en la ejecución es manejada y publicada. Cuando pequeñas caídas en la ejecución del plan se producen, acciones correctivas son requeridas.
7. Puesto que medidas no simples pueden representar adecuadamente un producto complejo, las medidas de calidad son tratadas como indicadores del rendimiento (performance). Esos indicadores son validados cuando sea posible a través del involucramiento temprano del usuario, como también por testeos operacionales simulados o actuales.

En la implementación de tal programa, un plan de calidad es producido durante el ciclo de planificación inicial del proyecto. El plan es documentado, revisado, manejado y comparado previamente con la experiencia actual.

Basado en esas comparaciones, una de las tres siguientes situaciones ocurrirá:

1. El resultado actual será conducido justo alrededor de un plan. Como el plan encontrará al objetivo del administrador y los objetivos son adecuadamente ofensivos, este es el resultado óptimo.
2. Los resultados actuales son significativamente peores que el plan. Esto significa que tampoco el plan fue más ofensivo que las capacidades de las organizaciones comunes. O la organización no lo está haciendo tan bien como es debido. En uno u otro caso el proyecto debe establecer un plan de acción para llevar la performance, tan cerrada al plan original como

práctica. El plan revisado es luego vuelto a ver y aprobado como antes en ningún caso el proyecto meramente cambia este plan sin incorporar acciones mejoradas significativamente.

3. Los resultados actuales son significativamente mejores que el plan. Desde un punto de vista motivacional este puede ser el peor caso. Por ejemplo, el equipo del proyecto puede estar operando seguro con un plan que ellos conocen que puede mejorar. Este puede ser conseguido por la revisión del plan para representar un cambio y luego establecer un plan de acción para encontrar esos nuevos objetivos.

Es importante, sin embargo, distinguir una situación de operación segura del equipo ofensivo que hace una verdadera y dramática mejora. Ellas merecen una celebración. En cada situación, sin embargo, cuando el rendimiento excede el plan, se requiere un nuevo y más agresivo plan.

El proceso de control de calidad no es nuevo, sino que ha sido usado exitosamente en muchos campos. Es también estrechamente paralelo el proceso de planificación del proyecto descrito en el Capítulo 6, en el cual los datos históricos son usados para asistir en hacer estimaciones de tamaño y recursos que son luego comparados con la experiencia actual. Después de unos pocos siglos del plan revisiones seguidas, y acciones de mejora, la organización aprenderá cómo hacer planes de calidad razonablemente exactos. El valor real de este proceso, sin embargo, es la habilidad para definir acciones de mejora de la calidad del software crecientemente efectivas. Buena planificación aún deriva de una apreciación de los esfuerzos que afectan el resultado. Con este conocimiento, uno es capaz de establecer formas progresivamente más efectivas de mejorar el proceso. La mejora progresiva, de paso, es un objetivo clave del programa de calidad.

5.1. Medidas de Calidad del Plan de Desarrollo.

Cuando el objetivo es motivar un superior rendimiento de desarrollo o mantención, los defectos son las medidas de calidad más prácticas. Esto es debido a que hay pocos otros datos que puedan ser razonablemente recolectados y usados durante el desarrollo. Además, si los defectos no son medidos, es difícil para los profesionales de software

tomar cualquier otra medida muy seriamente. Ellos saben por experiencia que el programa tiene defectos que deben ser identificados y fijados antes que el producto son liberados. Cualquier otra cosa tendrá así prioridad secundaria.

Las medidas de defectos específicos deben ser seleccionadas por cada organización basadas en los datos que pueden recolectar. Generalmente hay clases tales como las descritas en el Capítulo 15, pero el énfasis primario debe estar en aquellas que son probablemente causa de problemas al cliente. Muchas organizaciones restringen la definición de defectos válidos a aquellas que requieren cambios de códigos. Si se puede establecer criterios claros, sin embargo, los cambios de documentación también deben ser incluidos.

Una vez que los tipos de defectos han sido establecidos, la normalización para el tamaño del programa es generalmente requerida. Los defectos por 1000 líneas de código fuente es generalmente la más simple y más práctica medida para la mayoría de las organizaciones. Esta medida, sin embargo, requiere que la definición de "línea de código" sea establecida (de nueve como se estableció en el Capítulo 15).

El próximo aspecto es determinar que defectos medir y sobre qué periodo de tiempo. Esto de nuevo depende de los objetivos del programa de calidad. Si el foco está en administrar el proceso de desarrollo, medidas son necesarias durante el desarrollo, prueba y uso del cliente. Las medidas de desarrollo proveen un indicador oportuno de rendimiento, las medidas de la prueba luego proveen una validación temprana, y los datos de uso del cliente completan la evaluación de la calidad. Con este espectro completo de datos es posible calibrar la efectividad del desarrollo y test para largo plazo durante el uso del cliente y algunos medios identificar cada defecto con sus puntos de introducción. Los errores pueden entonces ser reparados y aquellos causados por la actividad de mantención pueden ser distinguidos.

Cuando se hace tal seguimiento a largo plazo, es posible evaluar muchas actividades del proceso de software. Por seguimiento, la inspección e historia del test del producto completo, por ejemplo, es posible ver cuán efectivas son cada una de estas acciones para encontrar y remover los defectos de los productos. Esta evaluación puede ser particularmente poderosa en el nivel de módulo, donde provee una forma objetiva de comparar la efectividad de la tarea usando los métodos de análisis estadísticos descritos en el Capítulo previo.

5.2. Retratando la Calidad del Software durante la Prueba de Aceptación y Uso.

Por seguimiento de los defectos encontrados por el cliente, Inglis sugiere un conjunto de curvas de defecto acumulativas, comenzando con la prueba del sistema. Estas curvas son normalizadas por 1000 líneas de código y graficadas como se muestra en la Figura 16.1

Aquí, los números acumulativos de defectos son graficados cada mes como se reciben. Para un producto con tres versiones, como en la Figura 16.1, la actual experiencia de la versión 1 es graficada en el tope, con el objetivo acumulativo de la vida del producto (LOP) mostrado a la derecha. El plan actual para fechar cada una de las versiones siguientes se muestran abajo. Para la versión 2, un plan de acción fue requerido en el mes 12, como es indicado por el círculo X. Este mismo dato puede ser graficado como una razón de defecto mensual, como se muestra en la Figura 16.2.

Aquí, los defectos encontrados cada mes son mostrados para el plan de la versión 2 y para la experiencia actual. Cuando es normalizada por tamaño del programa, tales datos históricos pueden proveer una referencia útil de producir nuevas estimaciones de calidad.

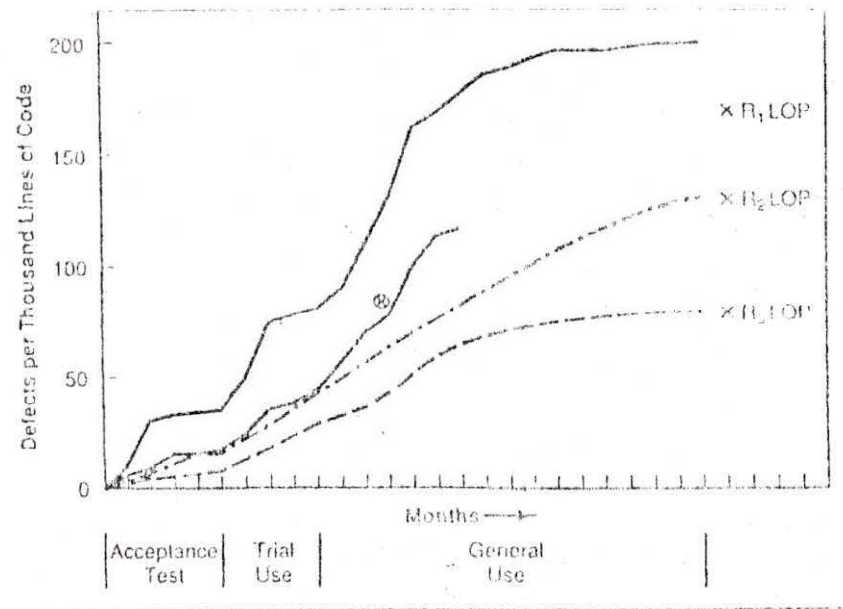


FIGURE 16.1
Plan Versus Actual - Three Product Releases

6. Estimando la calidad del Software.

Hacer una estimación de la calidad del software es importante recordar que cada proyecto es diferente. Mientras las estimaciones deben estar basadas en experiencias históricas, una buena estimación también requiere un entendimiento intuitivo de las características especiales del producto involucrado. Algunos ejemplos de los factores a considerar son:

- ¿Cuál es la razón anticipada de la instalación del cliente para este tipo de producto?

Una alta razón de instalación generalmente causa una aguzada cima temprana en la razón de defectos con una rápida declinación subsecuente. Típicamente los programas son instalados más rápidamente cuando requieren una mínima conversión y cuando ellos no afectan la operación completa del sistema. Los compiladores y los programas utilizados son ejemplos comunes de productos rápidamente instalados.

- ¿Cuál es la historia del producto liberado?

Una versión subsecuente puede instalarse rápidamente si se corrige las diferencias serias en la liberación anterior. Esto, de paso, requiere que la experiencia más temprana con la nueva versión sea positiva. Si no, se puede obtener una mala reputación y ser pobremente aceptada.

- ¿Cuál es el plan de distribución?

¿ El producto será liberado a todos los que lo requieran inmediatamente, será la disponibilidad inicial limitada, o habrá un período de prueba preliminar?

- ¿Está establecido el sistema de servicio?

Mirando a menos la calidad del producto, los usuarios están motivados y serán capaces de presentar reportes de defectos. Si no, el dato defectuoso no será suficientemente confiable para validar el proceso de desarrollo.

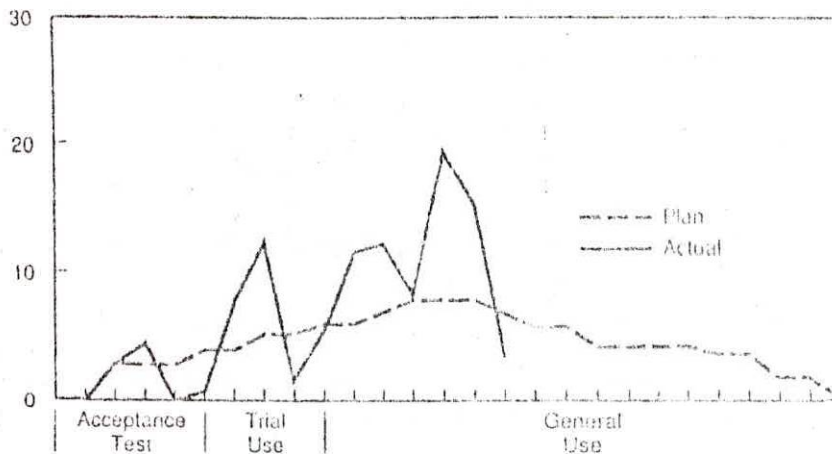


FIGURE 16.2
Defect Rates-Release 2

6.1 Haciendo una Estimación de la Calidad del Software.

Una estimación de la calidad de un programa se hace como sigue:

1. Programas recientes completados por la organización son revisados para identificar los más similares al producto propuesto. Donde el dato garantiza, esto es hecho para cada elemento del producto.
2. Datos de calidad disponibles sobre estos programas son examinados para establecer una base para estimar la calidad.
3. El producto significativo y las diferencias del proceso son entonces determinadas y sus efectos potenciales estimados.
4. Basado en estos datos históricos y los cambios planificados del proceso, se hace una proyección anticipada de la calidad para el proceso de desarrollo del nuevo producto.
5. Esta proyección es entonces comparada con los objetivos y necesidades de mejora del proceso, son ideadas para satisfacer los objetivos.

6. El perfil de la calidad del proyecto es entonces determinada para determinar las áreas para mejora potencial, y un perfil de la calidad deseada es producida.
7. Un plan de desarrollo es producido, que especifica el proceso a ser usado para obtener este perfil de calidad.

6.2 Modelos de la Calidad del Software.

Para hacer una estimación de la calidad adecuada, es esencial tener un modelo de la calidad. Mientras esta necesidad no sea un modelo matemático explícito, debe identificar las suposiciones básicas detrás de las estimaciones. La razón que los modelos de software mas metamáticos son no útiles para propósitos de estimación es que ellos hacen algunas suposiciones altamente restrictivas acerca del proceso de software. Mientras tales suposiciones son generalmente necesarias para hacer seguible matemáticamente el problema, ellas conducen a problemas serios cuando se trata de hacer proyecciones exactas.

Ejemplos de alguna de estas suposiciones típicas son:

- Las fallas son independientes
- El número de fallas es constante
- Cada falla es reparada antes que continúe el testeo
- Todas las fallas son observadas
- El testeo es de intensidad uniforme y representativo del ambiente operacional
- La razón de falla en cualquier momento es proporcional al número actual de defectos remanentes en el programa.
- Los tiempos entre fallas son independientes
- Los tipos de errores diferentes son de igual importancia
- Cada error exhibe la misma razón de falla
- Ningún error será introducido durante el testeo y proceso de reparación
- La razón de falla declinará durante el depurado y operación

Ya que estas suposiciones son claramente inapropiadas para la mayoría de los proyectos, no es sorprendente que grupos de desarrollo no usen modelos de la calidad en un trabajo.

6.3. Modelos de Calidad Intuitivos

Uno de los desafíos especiales de la ingeniería de software es que es un proceso intelectual que produce artefactos que no obedecen las leyes de la naturaleza. Los modelos necesarios para propósitos de estimación deben reflejar la forma en que la gente actualmente escribe programas. El hecho que tales modelos aún no han sido reducidos a formas matemáticas pulcras, no es un problema serio debido a que estamos tratando de construir un mecanismo algorítmico para generar estimaciones. Esta es una característica que hace tratable el proceso de ingeniería de software.

Algunas de estas características son:

1. La calidad del módulo del programa debe variar, con unos relativamente pocos módulos conteniendo es grueso de los errores.
2. Los módulos restantes probablemente contendrán unos pocos defectos distribuidos
3. La distribución de tipos de defectos también será altamente analizada, con unos pocos tipos cubriendo una gran proporción de los defectos.
4. Ya que los cambios de programación son altamente propensos al error, todos los cambios deben ser vistos como fuentes potenciales de inyección de defectos.

Mientras esta caracterización no es calificada como un modelo en un sentido formal, provee un esquema para la planificación de la calidad. Cada organización debe calcular su propia experiencia producida para determinar el grado en el cual éste modelo se ajusta a su situación. Es útil, por ejemplo, establecer el perfil de la distribución típica de Pareto de densidades de defectos del módulo, o tipos de defectos, como se muestra en la Figura 16.3.

Esta distribución de Pareto es una clasificación del número de defectos por causa, comenzando con el más prevaeciente a la izquierda. En estos datos desde cuatro programas Hewlett-Packard, sobre un tercio de los defectos estaban en las tres categorías más altas. Localizando sus esfuerzos de mejora del proceso en las categorías de defectos

prevalecientes de la Tabla 16.3, Hewlett-Packard fue capaz de obtener mejoras de calidad significativas.

TABLE 16.3

CATEGORIES OF SOFTWARE DEFECTS (9)

User Interface/Interaction

1. User needs additional data fields
2. Existing data needs to be organized/presented differently
3. Edits on data values are too restrictive
4. Edits on data are too loose
5. Inadequate system controls or audit trails
6. Unclear instructions or responses
7. New function or different processing required

Programming Defect

1. Data incorrectly or inconsistently defined
2. Initialization problems
3. Database processing incorrect
4. Screen processing incorrect
5. Incorrect language instruction
6. Incorrect parameter passing
7. Unanticipated error condition
8. Operating system file handling incorrect
9. Incorrect program control flow
10. Incorrect processing logic or algorithm
11. Processing requirement overlooked or not defined
12. Changes required to conform to standards

Operating Environment

1. Terminal differences
2. Printer differences
3. Different versions of systems software
4. Incorrect JCL
5. Incorrect account structure or capabilities
6. Unforeseen local system requirements
7. Prototyping language problem

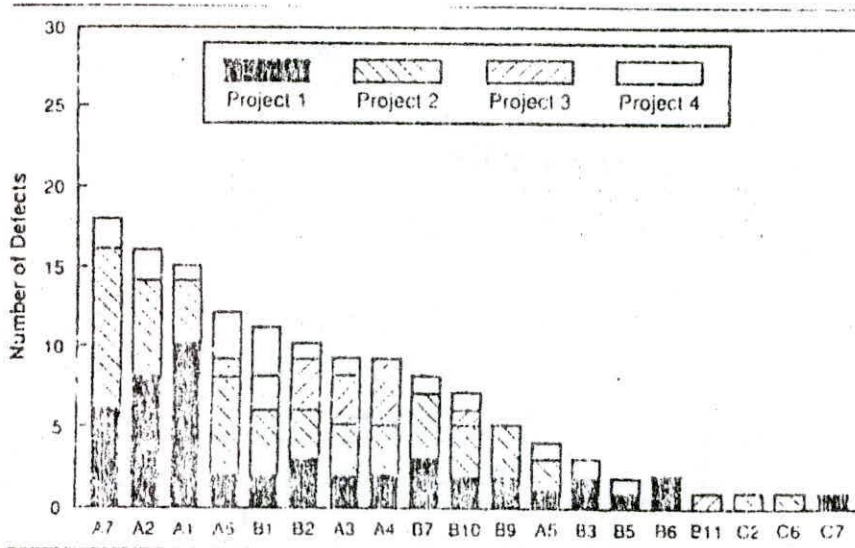


FIGURE 16.3
Pareto Analysis of Defects (9)

6.4. Estimación de la Calidad.

Al comparar los componentes y módulos del nuevo producto con elementos similares, cada característica de diseño e implementación únicas del programa deben ser consideradas. Como una consecuencia los perfiles de calidad para cada programa también será probablemente diferente. Un ejemplo hipotético de un perfil de defecto es mostrado en la Tabla 16.4

Aquí, las eficiencias de la inyección y remoción del defecto son calculadas comenzando con la fase de diseño de alto nivel (HLD) en la columna izquierda. El número anticipado de errores a ser inyectado es entrada, junto con el número esperado a ser encontrado durante el HLD, chequeo en el escritorio, e inspección HLD. EL número remanente es entrada, junto con la razón de inyección y la eficiencia de remoción. Esto es hecho para cada fase, y la eficiencia de remoción se refiere al porcentaje de todos los errores presentes en el punto que son removidos en esta fase.

La figura final en la parte de abajo de la Tabla 16.4 es la eficiencia de inspección. Esto es el porcentaje de aquellos defectos encontrados durante el desarrollo que son encontrados por inspección. Este es un buen indicador del proceso de desarrollo de alta calidad. El porcentaje 63.8

que figura mostrado aquí es respetable, pero no particularmente impresionante. Las eficiencias de inspección bajo el 50% generalmente indican un proceso de desarrollo que es altamente orientado al test y probablemente no muy efectivo en la remoción de defectos. Una ventaja de usar la eficiencia de inspección como una medida de desarrollo es que el dato requerido está disponible al final del desarrollo y no requiere de un sistema de reporte de defectos para la fase de utilización.

TABLE 16.4
DEVELOPMENT DEFECT PROFILE*

Defects	HLD	DLD	Code	U/T	I/T	S/T	Usage
Residual	0	4	13	34	19	13	9
Injected	10	21	63	5	2	2	3
Removed	6	12	42	20	8	6	12
Remaining	4	13	34	19	13	9	
Injection rate	10	21	63	5	2	2	3
Removal efficiency	60%	48%	55%	51%	38%	40%	100%
Cumulative efficiency	60%	58%	64%	81%	87%	91%	100%
Inspection defects	6	18	60				
Development Defects	6	18	60	80	88	94	
Inspection efficiency						63.8%	

*Per 1000 LOC

HLD - high-level design

DLD - detailed level design (modula level)

U/T - unit test

I/T - integration test

S/T - system test

injection rate - errors injected in the phase por 100 LOC

Removal efficiency for a process phase; $100 \times (\text{removed}) / (\text{residual} + \text{injected})$

Cuando los datos históricos en razones y eficiencias estan disponibles para una familia de programas desarrollados previamente, el estimador puede hacer juicios altamente informados acerca del próximo proyecto. El análisis de estos datos también sugerirá donde localizar el foco del esfuerzo del desarrollo para producir mejores resultados.

Revisando los perfiles de la calidad y decidiendo cuales son más representativas de los elementos producidos cuando se

está estimando, se puede hacer una proyección para el nuevo producto, suponiendo que nada hacambiano en el proceso. Ahora, la organización que esta haciendo el trabajo debe decidir que cambios harán en un proceso y cuáles serán probablemente la inyección resultante y las eficiencias de la remoción. Las nuevas razones son entonces estimadas y un plan es producido para el nuevo producto, como se muestra en las tablas 16.5 y 16.6 (La generación de estas figuras es descrita en la próxima sección). Ya que las razones mostradas son en defectos por 1000 líneas de código, las razones actuales deben ser factorizadas para representar el tamaño del programa planificado.

TABLE 16.5
PLANNED INJECTION RATES AND REMOVAL EFFICIENCIES

Defects	HLD	DLD	Code	U/T	I/T	S/T	Usage
Prior Project:							
Injection rate	10	21	63	5	2	2	3
Removal efficiency	60%	48%	55%	51%	38%	40%	100%
Cumulative efficiency	60%	58%	64%	81%	87%	91%	100%
New Project:							
Injection rate	10	21	63	1	1	0	1
Removal efficiency	60%	48%	70%	65%	41%	40%	100%
Cumulative efficiency	60%	58%	76%	92%	95%	97%	100%

Defects	HLD	DLD	Code	U/T	I/T	S/T	Usage
Residual	0	4	13	23	8	5	3
Injected	10	21	63	1	1	0	1
Removed	6	12	53	16	4	2	4
Remaining	4	13	23	8	5	3	0
Injection rate	10	21	63	1	1	0	1
Removalefficiency	60%	48%	70%	85%	41%	40 5	100%
Cumulativeefficiency	60%	58%	76%	92%	95%	97%	100%
Inspection defects	6	18	71				
Development defects	6	18	71	87	91	93	
Inspection efficiency						76.5%	

7. Eficiencia en la Remoción.

Una indicación de la eficiencia completa del proceso de desarrollo es la eficiencia de la remoción acumulativa, que indica el porcentaje acumulativo de los errores inyectados previamente que han sido removidos por el fin de cada fase del en forma gruesa con cada fase del proyecto, se debe enfocar la atención a remociones tempranas. Comparando las tablas 16.5 y 16.6, se puede ver que se supone que el nuevo producto tenga una calidad sustancialmente más alta que su predecesor. Esto fue cumplido en este caso mejorando la efectividad de

las inspecciones del código y el test de las unidades. Esto resulta entonces en una creciente eficiencia de la inspección.

Es interesante notar que una mejora de un 20% en la eficiencia de la inspección (es 63.8% a 76.3%) ha resultado en una mejora de tres veces en la calidad del producto liberado resultante (91% a 97%, ó 9% versus 3% de los errores a la izquierda en el producto).

8. Objetivos de Calidad.

Los objetivos de calidad de toda organización de desarrollo, de manufactura, o de servicios, deben ser establecidos por administradores experimentados. Estos administradores (senior) son los únicos que pueden establecer tales medidas estándares, y ellos son los que mejor aprecian los completos beneficios de un programa de calidad agresivo. Esto significa que los administradores deben establecer los objetivos de calidad en forma muy clara a su personal. Sin esta guía, los gerentes de productos le darán primera prioridad a la planificación e items de costos, porque allí estará el interés de su enfoque en general.

Mientras que cada organización debe establecer sus propios objetivos de calidad, un razonable punto de partida será:

1. Cada producto nuevo debe tener una calidad mejor que su predecesor.
2. La organización corporativa de calidad, con la ayuda de los grupos de software, establecerán las medidas de calidad que se usarán.
3. Cada administrador de un producto es responsable por producir un plan de calidad documentado para alcanzar dichos objetivos.
4. El rendimiento de la calidad del producto se juzmará de acuerdo a:
 - El grado hasta el cual los planes de calidad muestran mejoras.
 - La efectividad de los planes de acción para dirigirse a áreas de logros deficientes.
 - El deseo de los administradores de productos de establecer objetivos más agresivos cuando el rendimiento excede al plan.

5. La organización de calidad informará periódicamente a los administradores (senior) acerca del producto y del rendimiento organizacional en contra de estos objetivos.

Los objetivos de calidad agresivos son difícil de enfrentar. Ellos demandan acciones extraordinarias, pero ellos también pueden producir resultados extraordinarios. La clave es medir a los administradores en cuanto hasta donde son capaces de establecer sus propios objetivos de calidad en forma agresiva, y cuan efectivamente responden ellos con planes de acción cuando el rendimiento falla. Si el énfasis está en alcanzar los objetivos, los gerentes de proyectos probablemente confeccionarán planes de calidad que son lo más seguro posibles. Tanto los objetivos como los planes deberían ser lo suficientemente agresivos, de tal forma que un porcentaje razonable de los productos logren el objetivo deseado y se pueda recurrir a una acción remedial.

9. Planes de Calidad.

El plan de calidad documenta las acciones de calidad que la dirección intenta implantar. Este incluye la derivación de medidas de calidad, la identificación de los cambios del proceso planificado, y los logros de calidad anticipados.

Algunos ejemplar de acciones de mejoramiento típicos que podrían haber producido los avances mostrados en las Tablas 16.4, 16.5 y 16.6 son:

- Mejorar la cobertura de inspección de código, desde un 39% al 100%. Esto se espera para mejorar la eficiencia de remoción de la fase de codificación desde el 55% al 70%, resultando en una eficiencia de remoción acumulativa de código producido, del 64% al 76%.
- Aumentar las pruebas de unidad de módulo (UT), para incluir pruebas del 100% de todos los parámetros, tanto dentro como fuera de los obstáculos que se esperan. Esto se logra para mejorar la eficiencia de remoción de las UT, desde el 51% al 65%, y la eficiencia acumulativa de las UT producidas, del 81% al 92%.
- Aumento de la inspección de los cambios de códigos, desde la cobertura del 35% al 100%. Con esto se espera reducir la probabilidad de cambios defectuosos desde un 25% al 5, resultando en una razón de inyección de error reducido, y una eficiencia de remoción del proceso completo, del 97% a la S/T producida.

Mientras estos números son hipotéticos, ellos no son irracionales. Aunque cada una de las reducciones se aprecian como relativamente modestas, en total ellas resultan en un factor de mejoramiento de 9 a 3 defectos por KLOC. También debería tomarse en cuenta que cada nivel de mejoramiento se hace progresivamente más difícil en la medida que la calidad del producto mejora. Un cambio del 99.7% en una acción de remoción acumulativa, puede parecer modesta, pero reduce errores residuales del 0.5 al 0.3, o el equivalente a un 40%. Como se puede apreciar, cambios relativamente pequeños del proceso, pueden resultar en enormes mejoras del producto.

Existen muchas acciones que pueden tomarse para mejorar la calidad, y las más efectivas generalmente están relacionadas con la prevención de defectos más que con la remoción. Tales acciones incluirían el uso de metodologías de diseño avanzado, un nuevo lenguaje de diseño o implementación, confeccionar prototipos en forma temprana, o planificación mejorada de requerimientos.

El plan de calidad debería considerar minuciosamente, tanto la remoción de defectos como las estrategias de prevención de errores. Los métodos de prevención se discuten en el **Capítulo 17**.

Al cuantificar los planes y al hacer estimaciones explícitas de la eficiencia de remoción en cada etapa del proceso, es posible, llevar una cuenta del rendimiento y determinar el impacto de cada cambio del proceso. El seguimiento (rastreo) no solo ayuda al equipo de desarrollo a mejorar su rendimiento, sino que también proporciona datos de planificación mejorada. Si, por ejemplo, se encuentra que la prueba de unidad de todos los parámetros resulta en una eficiencia de remoción del 75%, tal prueba podría incrementarse más ampliamente. Según se establece en el Capítulo 11, existe evidencia de que tales mejoramientos son posibles.

El modelo de un plan de calidad típico, junto con algunos comentarios explicativos de cada una de las secciones principales, se entrega en la Tabla 16.7

10. Seguimiento y Control de la Calidad de Software.

Los elementos críticos de un sistema de administración de calidad del software son:

1. Una autoridad responsable es nombrada para hacerse cargo de los datos de calidad, y del sistema de rastreo y reportes.

2. El rendimiento de la calidad es rastreada e informada a esta autoridad, durante tanto el desarrollo como la mantención.
3. Se establecen los recursos para validar los datos reportados, y para retenerlos en la base de datos del proceso.
4. El producto actual y los datos del rendimiento organizacional son periódicamente revisados, de acuerdo a este plan:
 - Los resultados son revisados inicialmente con una administración de línea responsable, y se resuelve cualquier discrepancia.
 - Se determina la comparación del rendimiento con los objetivos, y si resulta peor que el plan, la administración de línea prepara un plan de acción para ser revisado en conjunto con la administración superior.
 - Si el rendimiento es sustancialmente mejor que el plan, la administración del producto establece objetivos más agresivos.
5. El rendimiento de la calidad se publica, y un informe explicativo se proporciona a la administración superior.
6. El rendimiento general es revisado periódicamente con la administración superior.

El propósito del plan de calidad es motivar a la acción, no evaluar a las personas. Cuando los objetivos agresivos no se cumplen, los planes de acción se inician para mejorar el rendimiento. Esto es exactamente el resultado deseado. Al contrario, cuando una organización siempre consigue sus planes de calidad, no resulta tan motivada para desarrollar e implementar mejoras. Esto es probablemente un programa de calidad no efectivo, ya que siempre hay espacios para mejoramientos.

TABLA 16.7
QUALITY PLAN OUTLINE

Product Introduction

- A general product description, including product's intended market, use, and current competitive or replacement quality expectations.

- Objectives
- Planned Market
- Competitive/predecessor product quality

Product Plans

- The approach to be taken with the product, by whom, and when.
- Critical dates, responsibilities, and checkpoints.
- Release plans
- Distribution plan
- Service plans
- operating environment
-

Process Description

- The development and service processes to be used
- The development process
- Description of any process elements that deviate from, the organization standard
- Explanation of the reasons and anticipated results of any deviations
- The assurance process
- Identification of those process actions and product elements to be reviewed
- Definition of the sampling methods to be and the review techniques to be applied
- The service process
- Description of any process elements that deviate from the organization standard
- Explanation of the reasons and anticipated results of any deviations

Quality Plan

- The quality goals and plans for this product
- Quality goals
- Prior relevant quality experience
- Quality actions planned
- Quality commitments

Risks and Exposures

- A listing of the key risks and exposures and the actions to be taken to address them
-
- Risks and exposures
- Responsibilities

- Checkpoints

11. Resumen

La calidad del producto es una de las mejores medidas de un proyecto de desarrollo de software. Las medidas de calidad establecen un claro registro de los progresos, representa un vehículo poderoso para establecer objetivos, y proceen un esquema (marco) gráfico para la acción presente.

Para lograr en forma consistente un rendimiento superior, la administración debe establecer objetivos de calidad de mucho desafío, y debe esforsarse por lograrlos. Esto requiere de un compromiso de la dirección superior, objetivos de calidad numéricos, planes de calidad, seguimiento de la calidad, y revisiones por parte de la administración.

Las cinco clases de medidas de calidad son: desarrollo, producto, aceptación, uso, y reparación. Estas medidas se caracterizan por ser objetivas, por ser programadas, estar disponibles, ser representativas o controlables. Se desea que las medidas estén siempre disponibles, sean fáciles de reunir, representativas de las necesidades del cliente, y directamente controlable por el desarrollo a la mantención.

Durante la planificación inicial del aproyecto, un plan de calidad de producido. Este plan es documentado, revisado y comparado con la experiencia actual. Al cuantificar los planes y hacer estimaciones explicitas de eficiencia de remoción de defectos en cada etapa del proceso, es posible el seguimiento de la calidad, y de este modo determinar los efectos de calidad de cada mejoramiento en el proceso del software.

El propósito del plan de calidad es motivar a la acción; no evaluar a las personas. Cuando los objetivos agresivos no son logrados, los planes de acción se inician para mejorar el rendimiento. Esto es exactamente el resultado deseado. Por el contrario, cuando una organización siempre logra sus planes de calidad, tiene poca motivación para desarrollar e implementar las mejoras. Estindica un programa de calidad poco efectivo, pues se pueden alcanzar mejoras.

FAVOR, DE VER EL LIBRO

ULTIMA FECHA