



Universidad Nacional del Litoral
**FACULTAD DE INGENIERÍA
Y CIENCIAS HÍDRICAS**

Ingeniería en Informática

Ingeniería de Software I

TEMA IV – El modelo de procesos estructurado

Introducción

Por muchas razones la captura de los requerimientos es la parte más dificultosa del desarrollo de un sistema de software. No se trata simplemente de la dificultad técnica del trabajo, si bien muchos proyectos exigen que el ingeniero tenga conocimientos profundos de la tecnología actualizada de procesamiento de datos. No se trata simplemente de las dificultades políticas que se presentan, especialmente en grandes proyectos, donde el nuevo sistema deberá servir a varios grupos de interés, posiblemente en conflicto. Tampoco se trata solamente de los problemas de comunicación que surgen en cualquier situación en la cual personas de diferentes antecedentes, con distintos enfoques del contexto y diferentes vocabularios deban trabajar juntas. Es la resultante de estas dificultades lo que hace que la definición de los requerimientos sea tan dura y exigente: el hecho es que el ingeniero debe hacer de intermediario entre la comunidad de los usuarios (aquellos que tienen la sensibilidad de sus problemas, pero encuentran dificultoso explicarlo) y el equipo técnico de desarrollo. El ingeniero debe hacer un balance entre aquello que es actualmente posible tecnológicamente y aquello que vale la pena hacer en la organización tal como está dirigida por sus administradores.

Si se hace un balance que resulte aceptable por todas las partes y que pueda resistir la prueba del tiempo, se ha logrado la parte más ardua del esfuerzo; si ha sido bien hecho, cualesquiera sean las dificultades de las distintas actividades de desarrollo, el sistema que se construya servirá a las necesidades de la organización. Si ha sido hecho pobremente, cualquiera sea la excelencia del software desarrollado, el sistema no satisfará las necesidades de la organización y fracasará en su propósito.

4.1 Los problemas iniciales del análisis

- El ingeniero suele encontrar muy difícil aprender lo suficiente de la organización para poder ver los requerimientos del sistema a través de los ojos del usuario. El temor es en el sentido de que al final puede haberse hecho un sistema técnicamente excelente pero no era lo que el usuario deseaba o necesitaba.
- Los usuarios suelen no saber lo que quieren o bien si lo saben con claridad, no lo transmiten de manera adecuada. Consecuentemente, el ingeniero debe de alguna manera interactuar con ellos y reflejarle de alguna forma lo que fue entendiendo en el avance del estudio. Suele entonces ser difícil mostrarle al usuario "la idea" de lo que hace actualmente el sistema, y de lo que hará. Las herramientas que se verán a continuación permitirán producir un modelo gráfico de un sistema que jugará el rol de esa "idea" constituyendo algo así como *los planos del sistema*.
- El ingeniero puede rápidamente verse abrumado por los detalles, tanto de la organización como por los detalles técnicos del nuevo sistema. La mayor parte del tiempo de la fase de captura de requerimientos, se emplea en obtener información detallada de la situación actual, los procedimientos administrativos, los documentos de entrada, los informes producidos y requeridos, las políticas en uso y los miles de hechos que surgen de una cosa tan compleja como una organización real. A menos que exista algún esquema o estructura para organizar estos detalles, el ingeniero se verá sobrecargado con hechos y papeles. Los detalles son necesarios y deberán estar disponibles cuando se los requiera por lo

que se deberá contar con herramientas para controlar los detalles.

- Los documentos donde se ubican los detalles del nuevo sistema (llamado de distintos nombres tales como: especificación de sistema, diseño general, especificación funcional, especificación de requerimientos, etc.) no son fácilmente comprensibles por los usuarios debido a su volumen y a eventualmente a algunos conceptos técnicos que contiene. En consecuencia, suelen decir que están de acuerdo con lo descrito. A la hora de entregar el sistema terminado (sobre todo si se utiliza un ciclo de vida en cascada), tendrán algo que sí pueden entender y recién entonces podrán reaccionar. Demasiado tarde por supuesto.
- Si el documento de especificaciones pudiera escribirse de manera que tuviera sentido para los usuarios, quizá no sería de mucha utilidad para los diseñadores físicos y programadores que deben construir el sistema. Suele entonces comenzarse a especificar el diseño físico antes que el modelo lógico del sistema este terminado, dando como resultado un sistema de baja calidad.

Aun contando con las mejores herramientas analíticas posibles, aparecerán algunos de los problemas enunciados. No existen herramientas analíticas que permitan al ingeniero saber lo que tiene en su mente el usuario si éste no se lo transmite. No obstante mediante la aplicación de ciertas herramientas lógicas, los problemas de análisis se verán facilitados en gran medida.

Puede ser difícil entender (y demostrarle al usuario que se entendió) por completo un proceso de la organización a través de una descripción verbal solamente; éste es el motivo fundamental por el que se recurre a una forma de representación gráfica que los ingenieros y los usuarios puedan comprender y leer fácilmente y que además ayudan a ilustrar los componentes esenciales de un proceso y la forma en la que interactúan.

Concretamente, se tratará una forma de mostrar a los usuarios un modelo tangible vívido del sistema, de forma tal que este grupo de usuarios se imagine lo que podrá hacer cuando el sistema esté en operación. En este tema se desarrollará una metodología que permita realizar diagramas para el modelado de las funciones o procesos que se dan dentro del sistema. La nomenclatura que se desarrolla es la clásica de los autores *Chris Gane & Trish Sarson*.

4.2 Modelo de análisis de procesos (*Process Analyst Models*)

La característica fundamental de los PAM es que no muestran los procesos en serie, vale decir que un PAM no comienza ni termina jamás. Existen tareas en una organización, que pueden ser efectuadas en paralelo y no una después de la otra presentando una secuencia. Los PAM, permiten reflejar perfectamente tales situaciones. La terminología modelo de análisis de procesos es utilizada en función del nombre que adoptan estos modelos en las más conocidas herramientas CASE. No obstante, el nombre con el que se ha difundido este tipo de gráficos es el de DFD (diagrama de flujo de datos).

Como su nombre lo sugiere, los diagramas de flujo de datos se concentran en los datos que se mueven a través del sistema y no en los dispositivos o equipos. Por medio del proceso del entendimiento del área de aplicación, el ingeniero identifica y describe los

datos que se mueven a través del sistema, por qué se introducen o se retiran y qué proceso se está llevando a cabo con ellos. Es muy importante determinar cuando entran o se retiran los datos del área de aplicación. Algunas veces los datos se almacenan para su uso posterior o su consulta en almacenamientos previos.

4.2.1 Ventajas del método

Estas notaciones sencillas, son fácilmente entendidas por los usuarios y personas de la organización que son parte del proceso que se estudia; por lo tanto, el ingeniero puede trabajar con los usuarios y realmente hacer que participen en el estudio del flujo de datos. Los usuarios hacen sugerencias sobre modificaciones de los diagramas para describir en forma más exacta la actividad de la organización; también pueden examinar los diagramas y señalar los problemas rápidamente, de manera que se puedan corregir antes de que se siga avanzando en el estudio. Si los problemas no se localizan en forma anticipada en el proceso de desarrollo, será muy difícil corregirlos posteriormente; incluso, evitarlos a tiempo, puede prevenir una falla del sistema.

El análisis del flujo de datos, permite aislar las áreas de interés de la organización, estudiarlas y examinar los datos que entran a los procesos y ver cómo se modifican cuando salen del mismo. Conforme se recopilan hechos y detalles, su creciente conocimiento de los procesos lleva al ingeniero a realizar preguntas sobre partes más específicas de ellos, que a su vez los conduce a una investigación adicional.

Una investigación de sistemas elaborada en forma amplia, produce conjuntos de muchos DFDs, algunos de los cuales proporcionan una visión general de los procesos principales, y otros dan un mayor detalle para señalar elementos de datos, almacenamiento de éstos y etapas del proceso para componentes específicos de un sistema mayor. Si el ingeniero desea revisar posteriormente el sistema en su totalidad, utiliza los diagramas generales y si está interesado en estudiar un proceso en particular, utilizará los diagramas de los procesos de menor nivel.

Los niveles de los diagramas pueden ser comparados con los mapas de carreteras y calles que se utilizan cuando se viaja en un área que no es conocida. En un viaje largo, primero se usa un mapa tal vez de nivel nacional, que indica las principales carreteras y ciudades. Conforme se acerca al lugar que quiere visitar, se necesitará seguramente un mapa más detallado que indique las rutas provinciales y caminos secundarios que permitan recorrer circuitos turísticos. Una vez que han arribado a la ciudad en la que se detendrán, un mapa de nivel nacional no les servirá para ubicarse dentro de esa localidad. Tampoco lo hará el mapa más específico con circuitos turísticos. Será necesario un plano detallado de la ciudad con el nombres de las calles y en el que se destaquen los lugares interesantes de conocer como sitios históricos, museos, centros comerciales y que contengan además, las indicaciones de cómo llegar a ellos. Esta información es esencial para encontrar las direcciones correctas de las calles; no obstante, no es útil cuando se inicia el viaje y se empieza a tener problemas para obtener una orientación. Los DFDs se utilizan de la misma forma. Se desarrollan y usan de manera progresiva, que va de lo general a lo particular para el sistema de interés.

Los DFDs muestran las características lógicas de las aplicaciones, es decir, señalan **qué** ocurre sin especificar el cómo. En otras palabras, los detalles físicos como

métodos de almacenamiento, dispositivos de entrada o componentes informáticos, no son aspectos principales para el ingeniero durante la parte inicial del desarrollo; sólo después de entender los componentes lógicos, deben estudiarse los componentes físicos.

4.2.2 Desventajas del método

El mayor inconveniente presentado, es que poco se indica acerca de los datos que participan. No dicen nada acerca del tipo que son, qué asociaciones lógicas se establecen entre ellos, cómo se vinculan, cómo se afectan, etc. En estos gráficos, lo fundamental es la representación de los procesos que acontecen dentro del sistema sin importar la naturaleza de los datos participantes. Para ello son utilizadas otras herramientas de modelado de datos que se representarán con otros tipos de modelos.

4.3 Objetos del modelo

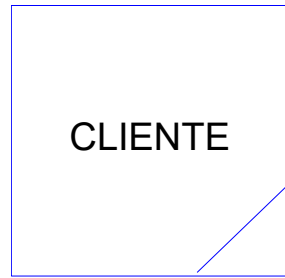
4.3.1 Las entidades externas

Las entidades externas son clases lógicas de cosas o de personas, las cuales representan una fuente o destino de datos. Por ejemplo: clientes, empleados, proveedores, contribuyentes, etc. También pueden ser una fuente o destino específico, por ejemplo, departamento de Contabilidad, DGI, depósito, etc. De acuerdo a lo descrito en los sistemas automatizados de información integrados, es totalmente válido que la entrada de flujo de información al sistema que se está considerando, provenga desde la salida de otro sistema. En ese caso, ese otro sistema emisor de información es una entidad externa.

Una entidad externa puede simbolizarse con un cuadrado o rectángulo de líneas llenas (suelen utilizarse también rectángulos con el lado inferior y el derecho de doble ancho o sombreado). La entidad puede identificarse con una letra minúscula en el ángulo superior izquierdo como referencia. En el interior, debe aparecer el nombre de la entidad. Por ejemplo la entidad externa CLIENTE, quedaría representada por:



Suele ocurrir que por la topología presentada en el diagrama, se requiera repetir un símbolo representando a una entidad externa que ya ha sido definida; para ello se crea una entidad externa sinónimo cuya única diferencia consiste en una línea inclinada en el ángulo inferior derecho del rectángulo. Para el caso de repetir la entidad externa CLIENTE, el símbolo correspondiente será:

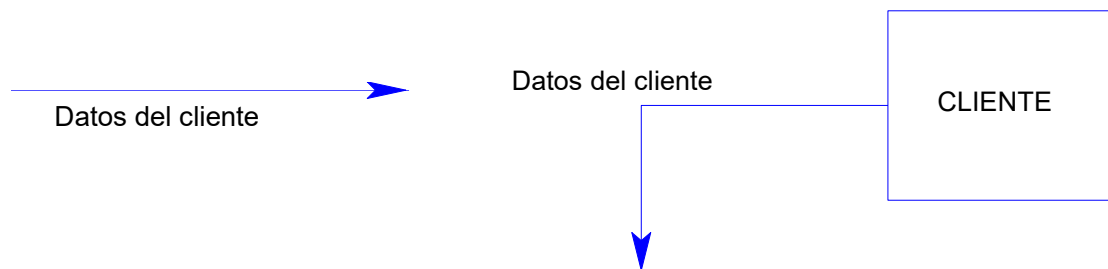


Mediante la designación del objeto entidad externa, se está estableciendo implícitamente que tal elemento se encuentra fuera de los límites del sistema considerado. A medida que se avance en el análisis y se aprenda más sobre los objetivos del usuario, se tomarán algunas entidades externas y se introducirán en el diagrama de flujo de datos del sistema, o por el contrario, se dejará de considerar alguna parte de las funciones del sistema designándola como una entidad externa con datos que fluyan desde y hacia ella.

4.3.2 Los flujos de datos

El flujo de datos se simboliza mediante una flecha, preferentemente horizontal y/o vertical, con la punta indicando la dirección del flujo. Para mayor claridad, y especialmente en los primeros borradores del diagrama, se utilizará una flecha con doble punta en lugar de dos flechas, cuando los flujos de datos estén apareados.

Los flujos de datos, se asemejan a una tubería por donde se envían paquetes de datos. Se puede hacer referencia a la línea del flujo de datos indicando los procesos, entidades o almacenes origen o destino, pero cada flujo de datos deberá tener indicada una descripción escrita de su contenido. El formato entonces de una línea de flujo de datos será:



4.3.3 Los procesos

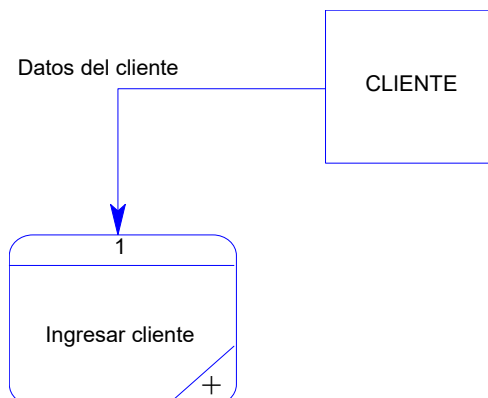
Los objetos mencionados hasta el momento han sido las entidades externas que pueden ser fuente o destino de datos los que necesariamente entrarán o saldrán del sistema a través de una tubería de flujo de datos. Ahora bien, para la existencia del flujo de datos –datos en movimiento– será necesaria la presencia de algún agente que impulse o aspire por alguno de los extremos de la tubería y de esa manera se produzca el movimiento. Los agentes encargados de establecer el flujo de datos (en cualquier sentido), son los denominados procesos.

Resulta necesario describir la función de cada proceso y para tener una fácil

referencia de qué acción realiza, se le asigna una única identificación (vinculada en lo posible, a un sistema físico o a un proceso del sistema operativo). Los procesos pueden simbolizarse con un rectángulo vertical, con los ángulos redondeados, dividido opcionalmente en tres áreas:

- La primera de ellas consiste en un número de identificación atravesado de izquierda a derecha en la sección superior del símbolo. No existe dificultad alguna en asignar numeraciones a los procesos si se tiene en cuenta que algunos serán explotados en otros procesos más especializados, lo que implicará la asignación de nuevos números. Otro tipo de ocurrencia puede ser que durante el trabajo se detecten procesos similares que seguramente se unificarán con la consecuente desaparición de números en la secuencia que inicialmente se planteó. Una vez asignada la identificación del proceso, ésta no deberá ser cambiada excepto –como se indicara- para aperturas o unificaciones, dado que se utiliza como referencia para los flujos de datos y para la descomposición del proceso en niveles inferiores. En caso de trabajar con alguna herramienta CASE, la asignación de los números de procesos será de forma automática y algunos, suelen tener utilidades que permiten la remuneración total controlando la consistencia del modelo completo.
- La segunda, corresponde a la descripción o etiqueta que permite a través de una frase breve, dar una idea de la funcionalidad del mismo. Esta descripción de la función deberá hacerse con una frase imperativa, que consistirá idealmente en un verbo en infinitivo seguido de una cláusula objeto (cuanto más simple mejor).
- La tercera es opcional. Algunas herramientas CASE no disponen de ella o bien la utilizan con otra finalidad. Según la teoría de los autores, esta área es utilizada para indicar el lugar físico en el que tal tarea o proceso es llevado a cabo (si se tratase de una oficina o repartición), o bien el nombre o identificación del programa que actualmente posee esa funcionalidad en caso que el sistema esté informatizado. Para la metodología de trabajo propuesta, la utilidad de esta zona se resume solamente a indicar si ese proceso posee o no niveles de descomposición representados con un signo de suma (+) en el extremo inferior derecho.

Para representar un proceso, se ha tomado como ejemplo la entidad externa CLIENTE, el flujo de datos Datos del cliente, y el proceso Ingresar cliente:



Nótese en el ejemplo, que la entidad externa CLIENTE pasa sus datos a través del conducto de flujo de datos Datos del cliente que es puesto en movimiento por el proceso Ingresar cliente. Es de destacar que el proceso mencionado posee descomposición, denotándose tal situación por la presencia del símbolo de suma (+) en el extremo inferior derecho del símbolo.

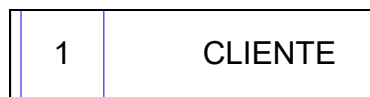
El ejemplo en cuestión es un absurdo, ya que un proceso responde a la definición de sistema, razón por la cual tendrá una o más entradas y necesariamente una o más salidas hacia otro destino. En la figura, el proceso Ingresar cliente no posee ninguna salida.

4.3.4 Los almacenes de datos

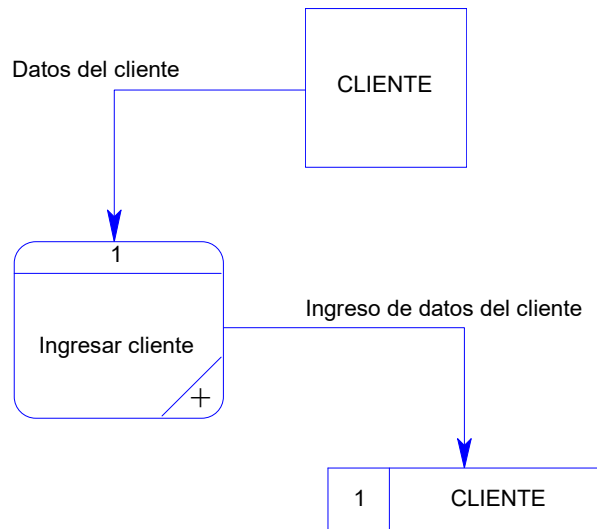
Sin tomar un compromiso físico durante el análisis, se encuentra que existen lugares donde resulta necesario definir datos para que puedan ser almacenados entre procesos. Estos lugares son los almacenes de datos. El símbolo que se adoptará para representar este objeto, consiste en dos líneas horizontales paralelas cerradas en el extremo izquierdo, preferentemente del ancho necesario para colocar el nombre descriptivo. Cada almacenamiento se indentificará con un número en un recuadro ubicado en el extremo izquierdo del gráfico para su fácil referencia. El nombre deberá ser lo suficientemente descriptivo para que el usuario note inmediatamente de qué se trata e interprete el contenido que posee.



Para evitar complicaciones con el cruce de líneas en un diagrama de flujo de datos, puede dibujarse el mismo almacenamiento de datos más de una vez en el diagrama, identificando los almacenes duplicados mediante líneas verticales adicionales colocadas a la izquierda como lo muestra la siguiente figura.



Cuando un proceso almacena datos, la flecha del flujo de datos se indica en la dirección al almacenamiento de datos. Cuando un almacenamiento de datos es accedido para ser leído únicamente, es suficiente con mostrar el grupo de elementos de datos recuperados como flujo de datos salientes. En definitiva, la posición de la punta de flecha de la línea de flujo, muestra el sentido del movimiento de los datos. A continuación se muestra cómo quedaría completo el proceso Ingresar cliente con ingreso de datos a partir de la entidad externa CLIENTE, el flujo de datos Datos del cliente e Ingreso de datos del cliente y el almacén CLIENTE.

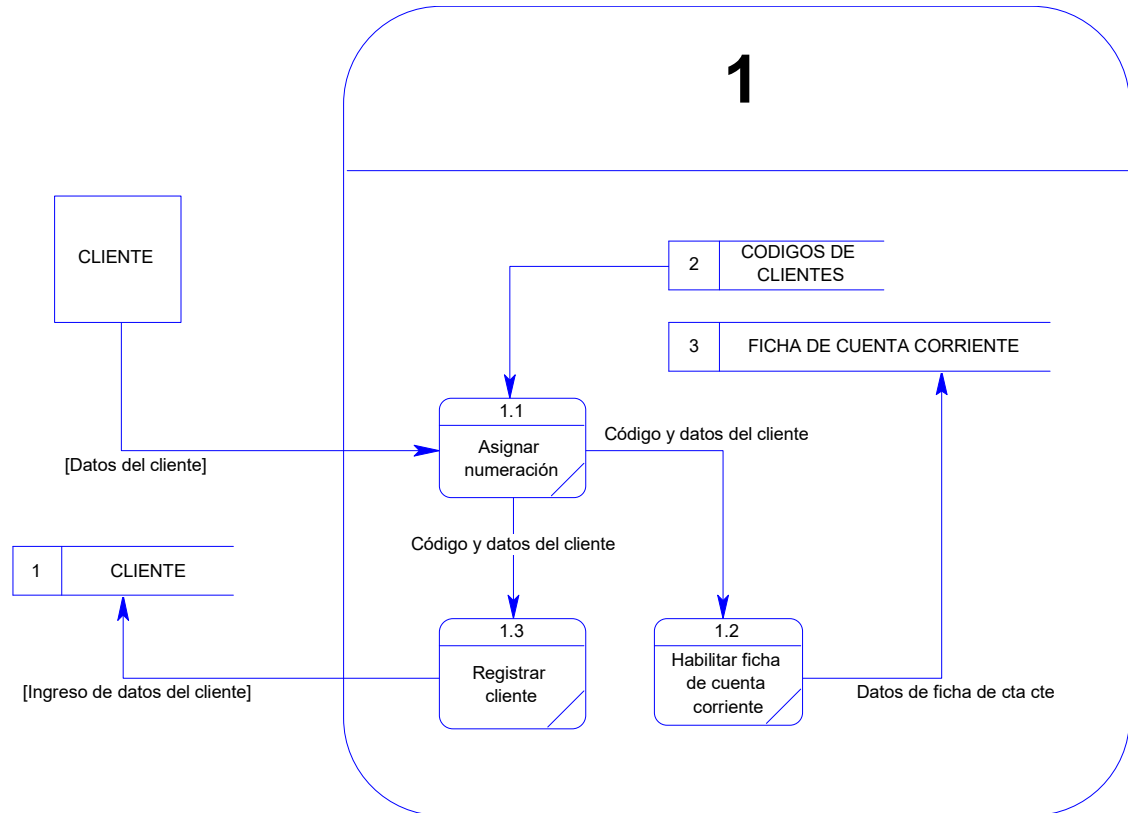


Nótese que el proceso toma los datos de la entidad externa y luego los ingresa en el almacén dispuesto para tal fin. Adviértase además que el proceso posee descomposición en procesos más sencillos o de menor nivel (signo +). Obsérvese además, que el diagrama en ningún lugar muestra cuáles ni de qué tipo son los datos del CLIENTE que se almacenan o ingresan. Esta situación es manejada por otro modelo distinto, no obstante es posible a través del uso de las herramientas de ingeniería de software asistido, la utilización de diccionarios de datos centralizados que pueden ser definidos en esta etapa.

4.4 La explosión de procesos

Un proceso de datos complejo, puede ser desagregado en otros procesos de menor jerarquía, pero que permitan reflejar con mayor claridad las tareas que se desarrollan. Tal procedimiento es denominado explosión de procesos, y su resultado es otro diagrama de flujo de datos. Cada proceso en el nivel inferior deberá estar relacionado, inversamente, con el proceso del nivel superior. Esto puede hacerse dándole a la casilla de proceso de nivel inferior un número de identificación que sea una fracción decimal de la casilla de proceso del nivel superior, por ejemplo, el 29 puede descomponerse en 29.1, 29.2, etc. y si es necesario llegar a un tercer nivel, 29.1.1, 29.1.2, etc.

La representación más clara de la explosión del proceso se obtiene dibujando los diagramas de flujo de datos de nivel inferior dentro de los límites que encuadran la casilla de proceso de nivel superior. Todos los flujos de datos entrantes y salientes de la casilla de proceso de nivel superior, deben entrar y salir a través del límite. Los flujos de datos que se muestran por primera vez en el nivel inferior, también pueden salir fuera de los límites. Por una cuestión organizativa y de nomenclatura, los almacenes de datos sólo se muestran dentro de los límites si son creados y accedidos por este proceso y no por otro.



Del ejemplo anterior, si se considera que el proceso Ingresar cliente puede ser descompuesto en:

1 Ingresar cliente:

- 1.1 Asignar numeración
- 1.2 Habilitar ficha de cuenta corriente
- 1.3 Registrar cliente

Cada uno de estos procesos, tienen su grupo de entradas y salidas todas interiores al proceso 1. Las entradas y salidas al y desde el proceso a explotar, son:

[Datos del cliente] entrada desde la *entidad externa* CLIENTE
 [Ingreso de datos del cliente] salida hacia el *almacén* CLIENTE

Nótese que han sido agregados los almacenes:

- 2 CODIGOS DE CLIENTES
- 3 FICHA DE CUENTA CORRIENTE

ya que solamente son accedidos dentro del diagrama de menor jerarquía. En el gráfico se representan las fronteras del proceso explotado y es de notar que la cantidad de entradas ([Datos del cliente]) y de salidas ([Ingreso de datos del cliente]) es la misma. Puede ocurrir que aparezcan más flechas de salida en el diagrama de menor nivel, pero que en el

nivel superior se corresponden con un agrupamiento lógico. Una situación similar puede presentarse con las entradas.

4.5 Desarrollo de los DFDs

Los primeros pasos en la determinación de requerimientos, como se ha indicado, se llevan a cabo para conocer las características generales de los procesos de la organización objeto del estudio. La parte de más alto nivel de los detalles, también se estudia. Conforme se comprenden mejor esos detalles, se profundiza y recopila más información específica y detallada. Se realizan preguntas específicas a un nivel de profundidad cada vez mayor: Este proceso se conoce como **análisis descendente** (*top-down*).

El paso de descripción de lo general a lo particular, se repite muchas veces en una investigación de sistemas: el entendimiento de un nivel de detalle, es extendido a un mayor detalle del siguiente nivel. En los grandes sistemas, un solo proceso, se puede extender muchas veces hasta que se describe una cantidad suficiente de precisiones para que de esta manera, pueda entenderse el proceso en su conjunto.

Los diagramas o modelos de análisis de procesos, no serán útiles si no se dibujan de forma apropiada o son difíciles de manejar. Cada nivel inferior se define utilizando de **tres a siete** procesos para explotar un proceso de mayor nivel. Utilizar más de siete, provoca que el diagrama sea difícil de manejar. Los diagramas son más fáciles de leer si la descripción ocupa una sola hoja de papel tipo A3. Si se necesitan más datos, el siguiente nivel puede extenderse.

4.5.1 Pautas para dibujar los diagramas de flujo de datos

1. Identificar las entidades externas involucradas. Ello implica decidir sobre un límite preliminar o fronteras del sistema. En caso de dudas, incluir dentro de los límites del sistema la primera "capa exterior" de los sistemas manuales o automatizados que tenga como interfaces.
2. Identificar las entradas y salidas programadas que se espera puedan producirse en el curso normal de la organización. Si la lista crece demasiado, seguramente existen agrupamientos lógicos de entradas y salidas.
3. Identificar consultas y pedidos de información que podrían surgir. Especificar un flujo de datos que defina la información "dada" al sistema y un segundo flujo de datos que indique que se "requiere" del sistema.
4. Jamás incluir decisiones en un DFD. De todas maneras, aunque se quiera, no se dispone de ningún mecanismo ni simbología para referenciarlas.
5. Duplicar entidades externas y almacenes de datos en caso de que se enmarañe demasiado el dibujo. Los diagramas deben resultar claros y deben evitarse los cruces de líneas de flujo de datos.
6. Producir una explosión de nivel inferior de cada proceso definido en otros diagramas, utilizando las convenciones especificadas anteriormente. Si es necesario, realizar modificaciones al diagrama de nivel superior.

4.6 Detalles de los componentes funcionales

Aunque el DFD, proporciona una visión global bastante conveniente de los componentes funcionales del sistema, no da detalles de éstos. Para mostrar detalles acerca de qué información se transforma y de cómo lo hace, se ocupan dos herramientas textuales de modelado adicionales: el diccionario de datos y la especificación de procesos.

4.6.1 El diccionario de datos

Cualquiera de los objetos que son creados en la construcción del modelo de análisis de procesos, sean estos *entidades externas*, *flujos de datos*, *almacenes o procesos*, deberá estar registrados en algún lugar. Para el ejemplo seguido, se debe conocer qué significa la *entidad* externa CLIENTE, qué significa el flujo de datos Datos del cliente y además, será necesario conocer cuáles son esos datos del cliente a que se hace referencia. Esta información deberá estar organizada en un documento al que se lo denomina Diccionario de Datos, los que hoy en día y mediante el uso de herramientas CASE, su mantenimiento se hace de manera automática. En general un *diccionario de datos* deberá incluir:

- **Entidades Externas**

- Identificativo de la entidad externa
- Nombre de la entidad externa
- Significado
- Ejemplo de ocurrencia

- **Flujos de datos**

- Identificativo del flujo de datos
- Nombre del flujo de datos
- Detalle de datos que transporta:

n ocurrencias

Nombre del dato
Tipo de dato
Significado
Ejemplo de ocurrencia

Origen (*entidad externa, proceso, almacén*)

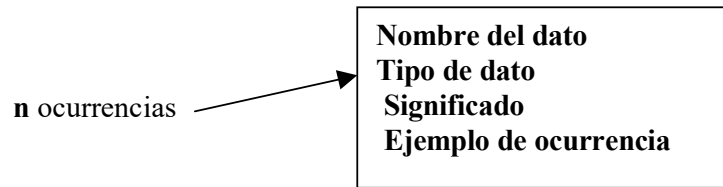
Destino (*entidad externa, proceso, almacén*)

- **Procesos**

- Identificativo de proceso
- Nombre del proceso
- Descripción

- **Almacén**

- Identificativo del almacén
- Nombre del almacén
- Descripción
- Detalle de datos que aloja:



Es de notar que tanto las líneas de *flujo* y los *almacenes*, manejan la misma información: **datos**, radicando su diferencia que en el primer caso están en movimiento y en el segundo, son estáticos. Hasta el momento no se han tratado los ítems de datos. De acuerdo a lo especificado, los atributos de los ítems de dato son:

- **Ítems de dato:**
 - Nombre del dato
 - Tipo de dato
 - Significado
 - Ejemplo de ocurrencia

Al manejar los datos como otro elemento más que es componente de cualquiera de los objetos que lo pueden contener o mover, no se hace necesario especificarlos junto con los *flujos* y con los *almacenes*, sino que se agrupan separadamente. Esto además permite identificar en principio todos aquellos datos que están repetidos pero que significan distintas cosas o bien aquellos que tienen distinto nombre pero significan lo mismo (caso más difícil de controlar).

En definitiva, un diccionario de datos, es una lista de todos los elementos incluidos en el conjunto de los modelos de análisis de procesos que describen un sistema. El diccionario de datos, almacena detalles y descripciones de estos elementos. Si se desea conocer cuantos caracteres hay en un dato, con qué otros nombre se le conoce en el sistema, o en dónde se utilizan dentro del sistema, el diccionario de datos- si está desarrollado apropiadamente-, debe proporcionar esa información. El diccionario de datos, se desarrolla durante el análisis de los procesos y el análisis conceptual complementándose entre ambos.

4.6.2 La especificación de procesos

Un modelo de análisis de procesos o diagrama de flujo de datos, no muestra ningún tipo de decisión y además, no muestra el detalle del proceso de último nivel. Para modelar las decisiones y describir acabadamente el funcionamiento de un proceso, se recurre a la Especificación de Proceso. Este tipo de herramienta, es la que le proporciona mayor claridad al proceso que deba automatizarse mediante un programa de computadora.

La especificación de procesos, tiene como propósito definir lo que debe hacerse para transformar entradas en salidas. Es una descripción detallada de la política de negocios del usuario que cada proceso lleva a cabo. Existen varias metodologías que consideran la especificación de procesos, pero, todas deben satisfacer dos requerimientos fundamentales:

- La especificación del proceso debe expresarse de una manera que puedan verificar tanto el usuario como el analista. Precisamente por esta razón se evita el lenguaje narrativo como herramienta de especificación: es notoriamente ambiguo, sobre todo si describe acciones alternativas (decisiones) y acciones repetitivas (ciclos). Por naturaleza, tiende a causar gran confusión cuando expresa condiciones booleanas compuestas (combinaciones de operadores *AND*, *NOT* y *OR*).
- El proceso debe especificarse en una forma que pueda ser comunicada efectivamente al público amplio que esté involucrado. A pesar de que el analista es típicamente quien escribe la especificación del proceso, habitualmente será un público bastante diverso de usuarios, administradores, auditores, personal de control de calidad y otros, el que leerá la especificación de proceso.

Existe una gran interacción y superposición entre la especificación de procesos y la explosión de procesos. Las especificaciones de proceso sólo se desarrollan para los procesos de más bajo nivel en un conjunto de diagramas por niveles en un modelo de análisis de procesos. Además, la especificación de proceso para un proceso de nivel superior, es el diagrama de análisis de procesos de nivel inferior. Escribir una especificación de proceso adicional, sería superfluo y redundante; esto es, crearía una especificación más difícil de actualizar.

La principal herramienta para la especificación de procesos - no implicando que sea la única- es la descripción mediante un lenguaje estructurado (pseudo código). Su propósito, es hacer un balance razonable entre la precisión del lenguaje formal de programación y la informalidad y legibilidad del lenguaje cotidiano. Una frase en lenguaje estructurado puede consistir en una ecuación algebraica, o en una sencilla frase imperativa que consista en verbo y un objeto. Los verbos deben escogerse de entre un pequeño grupo de verbos orientados a la acción tales como: conseguir (o aceptar o leer), poner (o mostrar o escribir), encontrar (o buscar o localizar), sumar, restar, multiplicar, dividir, calcular, borrar, validar, mover, reemplazar, fijar, ordenar.

En muchas organizaciones, se llega a la conclusión de que entre 40 y 50 verbos son suficientes para describir cualquier política dentro de una especificación de proceso. Los objetos (el tema de las frases imperativas sencillas) deben consistir sólo en datos que se han definido en el diccionario o ser términos locales. Los términos locales, son aquellos que se definen explícitamente en una especificación de proceso individual; sólo son conocidos, relevantes y con significado dentro de dicha especificación de proceso (por ejemplo un cálculo intermedio).

En el siguiente ejemplo, se muestra una sencilla aplicación de especificación de proceso, que no considera datos definidos en el diccionario, simplemente quiere darse una aproximación a la manera de emplear el método:

1 - SI el monto de una factura multiplicado por el número de semanas de retraso en el pago supera los 10.000 dólares **ENTONCES:**

- a. Proporcionar una fotocopia de la factura al encargado de ventas que llamará al cliente.
- b. Anotar en el reverso de la factura que se le dio una copia al vendedor, junto con la fecha en

- la que se hizo esto.
- c. Volver a archivar la factura para estudiarla de nuevo dentro de dos semanas.

2 - EN CASO CONTRARIO, SI se han enviado más de cuatro recordatorios, **ENTONCES:**

- a. Dar una copia de la factura al vendedor apropiado para que llame al cliente.
- b. Registrar en el reverso de la factura que una copia ha sido enviada al vendedor, y la fecha en la que se hizo esto.
- c. Volver a archivar la factura para reexaminarla dentro de una semana.

3 - EN CASO CONTRARIO (la situación aún no ha alcanzado proporciones serias):

- a. Añadir 1 al contador de avisos de mora registrado en el inverso de la factura (si no se ha registrado tal contador, escribir: "*cuenta vencida de avisos de mora = 1*").
- b. **SI** la factura archivada es ilegible, **ENTONCES** hacer una nueva.
- c. Enviar una copia de la factura al cliente, con el sello: "*n-ésimo aviso: pago de factura vencido, Favor de remitir inmediatamente*", donde n es el valor de avisos de mora.
- d. Registrar en el reverso de la factura la fecha en la que se envió el n-ésimo aviso de mora.
- e. Volver a archivar la factura para examinarla dentro de dos semanas.

4.7 Conclusión

Los sistemas siempre forman parte de sistemas mayores y siempre pueden dividirse en sistemas menores. Este punto es importante por dos razones:

- Sugiere una forma obvia de organizar un sistema computacional que se está estudiando para luego desarrollar, por el procedimiento de dividirlo en sistemas menores.
- Sugiere además que la definición del sistema que se está estudiando ha sido arbitraria. Se hubiera podido escoger un sistema ligeramente menor o mayor, dependiendo del lugar en el que se coloquen las fronteras y definan los alcances del mismo. Determinar lo que deberá abarcar un sistema y definirlo cuidadosamente para que todos los integrantes del equipo de desarrollo conozcan su contenido es una actividad crucial. Resulta una tarea muy difícil determinar drásticamente los límites. Tanto los usuarios como el equipo de trabajo, a menudo piensan que la frontera del sistema es fija e inmutable y que todo lo que se encuentre fuera, no merece la pena ser estudiado, aunque la realidad, muchas veces demuestra lo contrario. A la hora de definir ese límite o fronteras interviene siempre la naturaleza jerárquica de la complejidad.