

Módulo I: SUBSISTEMA SOPORTE DE DISEÑO DE PRODUCTO

Juan C. RAMOS¹, Lorena BEARZOTTI¹, Enrique MILANI¹, Federico WOSCOFF¹, Mariana GORSKY¹
María R. GALLI², Omar CHIOTTI³

(1) **GIDSATD - UTN** - Facultad Regional Santa Fe - Lavaisse 610 - 3000 SANTA FE - ARGENTINA

(2) **INTEC - CONICET**- Güemes 3450 - 3000 Santa Fe - e-mail: mrgalli@intec.edu.ar

(3) **INGAR - CONICET** - Avellaneda 3657 - 3000 Santa Fe -e-mail: chiotti@arcride.edu.ar

RESUMEN

El objetivo de nuestro Grupo de Investigación y Desarrollo es analizar, diseñar y desarrollar un prototipo generador de Sistemas Soporte de Decisiones (SSD) con capacidad de adaptación a diferentes estructuras organizativas de la empresa, con flexibilidad para incorporar cambios de los productos y de los procesos de producción y ofrecer soporte a todos los puntos de decisión de la organización.

En este trabajo presentamos una descripción del subsistema de soporte de la actividad de diseño de producto. Este subsistema es uno de los doce módulos que integran el prototipo, y uno de los cinco que actualmente están siendo desarrollados por nuestro grupo. En particular presentamos la estrategia diseñada para el manejo de la información asociada al producto y una descripción del análisis y diseño orientado a objetos del subsistema.

INTRODUCCIÓN

Los permanentes cambio tecnológicos y la nueva visión de las empresas tendientes a satisfacer las expectativas del cliente, resultan en una considerable reducción del ciclo de vida de los productos y de los procesos productivos. Esto obliga a las empresas a permanentes cambios de sus productos y de sus procesos. Bajo esta condiciones, el procesamiento de datos juega un rol fundamental al momento de tomar decisiones.

En los últimos años muchos trabajos han sido publicados describiendo Sistemas Soportes de Decisiones (a modo de ejemplo podemos citar: Parker y otros, 1994; Adamopoulos y otros 1994; Niehaus, 1995). La característica común de estos sistemas es que han sido diseñados para servir de soporte a las decisiones de un sector particular de la organización (scheduling de personal, planeamiento y scheduling de producción, management, planeamiento y control de inventarios, etc.), por esta razón, la estructura de estos sistemas se ajusta a la situación particular a la que sirven de soporte, y difícilmente puedan ser adaptadas a otras estructuras organizativas, aún dentro de la misma empresa. Sage, 1991, considera que esta es una importante limitación de un sistema soporte de decisiones, especialmente debido al dinamismo de las estructuras de las organizaciones actuales.

Nuestro Grupo está trabajando con el propósito de desarrollar un prototipo generador de SSD, el cual permita generar SSD para diferentes tipos de empresas de producción (alimenticias, textiles, manufactura, etc.), independientemente de su organización. Al respecto, en primer lugar hemos diseñado la Base de Modelos global del sistema (Rico y otros, 1997). Posteriormente hemos realizado el análisis y diseño de cinco de los doce módulos que conforman el sistema global. En este trabajo presentamos el *subsistema soporte de diseño de Producto*. En primer lugar describimos la estrategia utilizada para organizar la información de los productos y en segundo lugar presentamos un análisis y diseño orientado a objetos del módulo.

ESTRATEGIA DE ORGANIZACIÓN DE LA INFORMACIÓN

El objetivo de este subsistema es dar soporte a la tarea de diseño de producto, brindando información a cerca de los productos actuales y permitiendo incorporar nuevos diseños. En este trabajo, utilizamos el término *información de un producto* para referirnos no solo a las *características comerciales* del producto, sino también a la *lista de materiales* requeridos y el *conjunto de actividades* necesarias para su producción.

Como lo expresáramos en la introducción, hoy en día, las empresas ofrecen productos con cientos de variantes. Sería poco eficiente almacenar todas las posibles variantes en la Base de Datos debido a la redundancia de datos comunes de productos con ligeras variaciones. Erens y otros, 1994, consideran que una característica deseable de un sistema para soportar la actividad de diseño de productos es evitar la redundancia de datos. Para ello proponen organizar los productos en familias. La principal limitación de este trabajo es que sólo considera los datos comerciales del producto. Es decir, no tiene en cuenta la lista de materiales y el proceso de producción.

Para precisar mejor los conceptos anteriores presentamos dos ejemplos de productos. En base a estos ejemplos describiremos la estrategia desarrollada para organizar la información en el subsistema.

Consideremos los siguientes productos, especificados comercialmente:

Silla A: Asiento y respaldo tapizados en tela rústica roja. Estructura de caño de hierro sección circular, soldado y pintado con esmalte sintético negro. Respaldo alto.

Silla B: Asiento y respaldo tapizados en cuero natural. Estructura de hierro de sección cuadrada, atornillado y patinado en azul y oro. Respaldo bajo.

El Proceso de Producción de estos productos se describe esquemáticamente en la Figura 1:

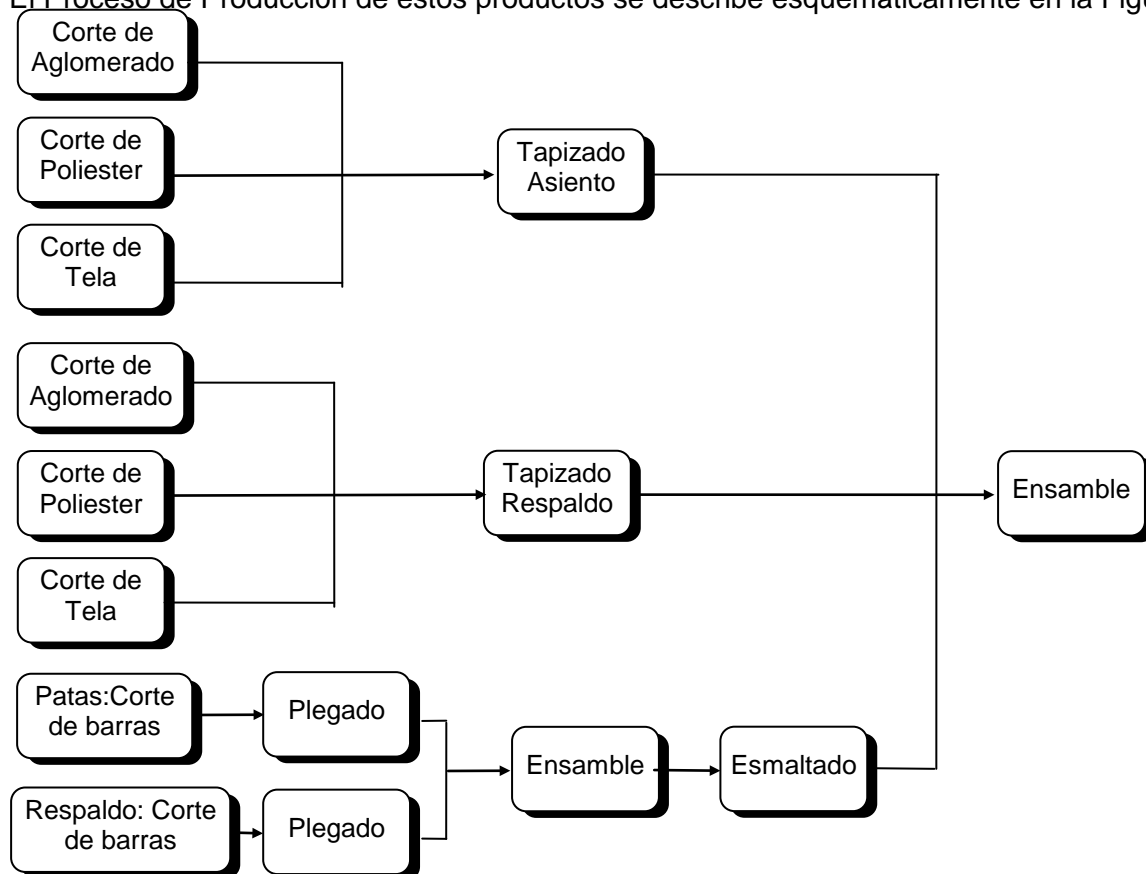


FIGURA 1: Representación esquemática del Proceso de Producción de las sillas A y B .

Analizando las sillas A y B, vemos en principio que se trata de dos productos diferenciados entre sí sólo en algunas características comerciales (tapizado, esmaltado, altura del respaldo, tipo de ensamble, etc.). Es fácil ver también que es posible generar cientos de productos distintos a partir de ligeras modificaciones de las características comerciales, como ser: el tipo de materiales (material de la estructura, tela de tapizado, esmalte, etc.), las dimensiones (alto del respaldo, etc.) y el tipo de ensamble (soldadura, remachado, atornillado, etc.).

Además de estas observaciones directas, es posible deducir que ligeras modificaciones en las características comerciales pueden también implicar cambios en las actividades de producción. Por ejemplo: El ensamblado de la estructura puede ser por soldadura, remachado, atornillado, etc. El esmaltado puede ser sintético, cromado, patinado, etc. Todas éstas son actividades diferentes, que requieren equipos, tiempos de operación, servicios, mano de obra y costos distintos. Toda esta información debiera estar disponible en forma explícita para el diseñador. Por esta razón, consideramos que un sistema de soporte a la tarea de diseño de producto, debe incorporar en forma explícita no sólo la descripción de las *características comerciales*, sino también la *lista de materiales* y el *proceso de producción*.

Para almacenar toda esta información, evitando la redundancia de datos, hemos desarrollado una estrategia que se basa en dividir el producto en *componentes* y expresar el proceso de producción en término de *actividades*. Un *componente* es considerado como una entidad independiente que requiere actividades, materiales y eventualmente otros componentes. Una *actividad*, definida siguiendo los conceptos de Philippe Lorino, 1993, es también una entidad independiente que requiere recursos (equipos y tiempo), materiales (materiales no incorporados al producto, por ejemplo solventes, papel de lijar, etc.), servicios (energía eléctrica, vapor, agua fría, etc.) y mano de obra.

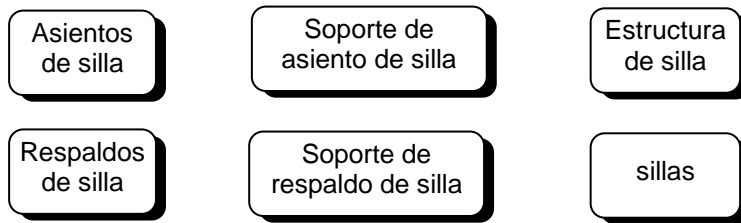
Es decir, desde el punto de vista de los datos, desaparece el concepto de producto y de proceso de producción, y por ende desaparece también la lista de materiales como tal. Luego, el producto, el proceso de producción y la lista de materiales es información que se genera mediante procedimientos (métodos).

Con esta conceptualización, analicemos ahora el ejemplo de las sillas:

Podemos considerar a la planta de producción de sillas como una entidad capaz de realizar las siguientes actividades:



Por otra parte, podemos considerar que la fábrica produce los siguientes componentes:



Analicemos algunos componentes en particular, por ejemplo el componente *estructura de silla* y el componente *silla*:

Estructura de silla [requiere]

1. actividades { Ensamble, Pintura }
2. materiales {Esmalte sintético, Antióxido, electrodos para soldadura }
3. componentes { Soporte de asiento (patas), Soporte de respaldo }

Silla [requiere]

1. actividades { Ensamble por atornillado, Embalaje }
2. materiales { Tornillos, Polietileno, Rótulo }
3. componentes { Asiento, Respaldo, Estructura de silla }

Queda así definida la estructura de la familia de las sillas. A partir de esta estructura, es fácil desarrollar un método que permita generar el proceso de producción asociado a un producto particular de la familia, y un método para generar la lista de materiales correspondiente. De este modo, nuevos diseños de sillas pueden ser incorporados al sistema a través de:

- ⇒ La incorporación de nuevos materiales (de tapicería, de soporte, esmaltes, etc.);
- ⇒ la definición de nuevas dimensiones (largo, ancho, alto, etc.);
- ⇒ la prestación de nuevas actividades por parte de la planta (nuevo sistema de recubrimiento de la superficie metálica, etc.)
- ⇒ la incorporación de nuevos componentes (incorporación de un componente estructural integral, el cual a diferencia del anterior no requiere de otros componentes)

Finalmente, cuando el diseñador o el cliente especifican un producto en particular, equivale a indicar en el sistema que elemento (instancia) particular de cada componente lo conforma y cuales son las características específicas de esa instancia (Ej. Tela: Pana sintética color rojo, Pintura: esmalte sintético color negro mate, Alto del respaldo: 72 cm., etc.).

Cabe destacar que la organización de la información asociada a las actividades que presta la planta, corresponde al subsistema *Capacidad de Producción*, el cual se encuentra en desarrollo en nuestro Grupo y será descripto en otro trabajo.

Los *materiales* también son descriptos por un cierto conjunto de información (unidades comerciales, especificaciones, disponibilidad, plazos de entrega, etc.). Esta información está a cargo del subsistema de *Planificación y Gestión de Materiales*, el cual se describe en otro trabajo.

Con esta breve descripción en base al ejemplo, hemos presentado la estructura desarrollada para representar la información de los productos en el subsistema de soporte a la actividad de

diseño de producto, evitando la redundancia de datos. En la sección siguiente describimos la estructura de clases del subsistema.

ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS DEL SUBSISTEMA

Describimos en esta sección la estructura de clases del subsistema. La misma fue desarrollada a través de un análisis y diseño del subsistema realizado siguiendo la metodología de objetos de G. Booch (1993). Para realizar el análisis y diseño del módulo, comenzamos definiendo todos los escenarios principales a los cuales el sistema debiera dar soporte. A modo de ejemplo y a efectos de introducir el problema presentamos el escenario principal *Incorporar un Nuevo Producto*.

ESCENARIO : *Incorporar Un Nuevo Producto*

1. El diseñador requiere incorporar la definición de un nuevo producto.
2. Define las características generales del producto.
3. Define las partes componentes del producto
4. Define los materiales necesarios para el producto.
5. Define las actividades necesarias para construir el producto.

De este escenario, haciendo análisis de sustantivos, surgen clases/objetos candidatos: *Producto*, partes *Componentes*, *Materiales*, y *Actividades*. Las “características generales” se intuyen como *atributos* de la clase candidata *producto*. En la figura 4 se muestra el Diagrama de Interacción correspondiente, el cual representa la interacción entre las clases/objetos candidatos.

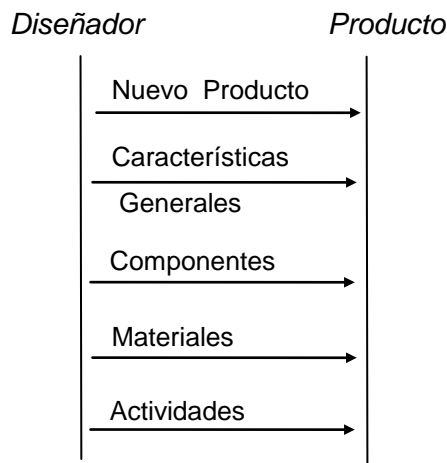


Figura 4: Diagrama de Interacción

La primera aproximación de las relaciones existentes entre las clases candidatas es de composición. Esto es *Producto* está compuesto por *Componentes*, *Actividades*, *Atributos* y *Materiales*. Esta relación se representa en el Diagrama de Clases 1.

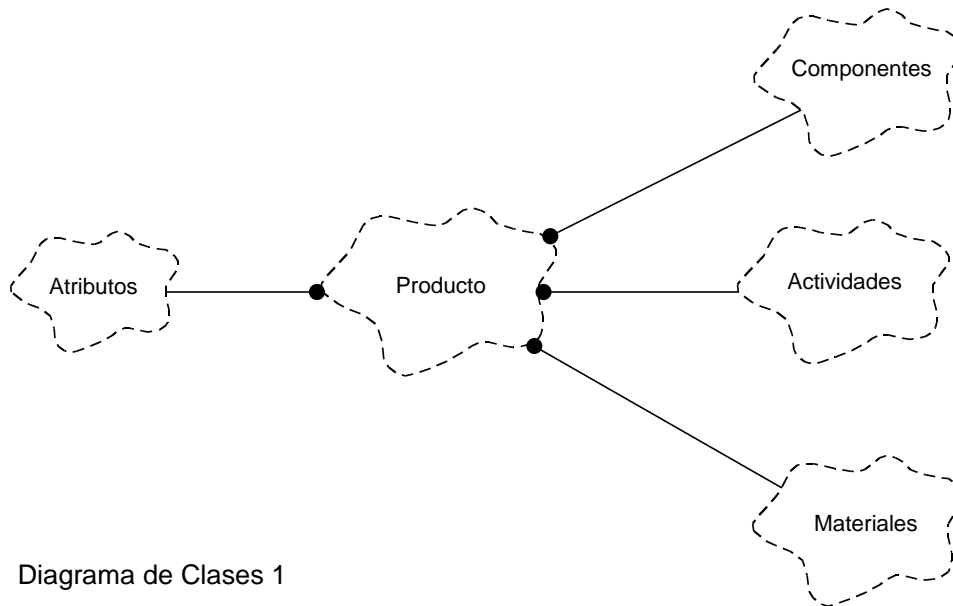


Diagrama de Clases 1

Debemos destacar aquí que los atributos han sido considerados como una clase y no como un atributo simple dentro de la definición de la clase, porque sabemos que los atributos requerirán cierta información complementaria (unidades de medida, valores actuales, etc.)

Analizando distintos productos, hemos encontrado que existen productos que denominamos *simples* los cuales se producen a partir de materiales primarios (para la empresa), mientras que existen productos que son generados a partir de otros productos, a éstos los denominamos *compuestos*. Como ejemplo podemos considerar una fábrica de ventiladores. Si adquiere los motores que utiliza para fabricar el ventilador, en éste caso consideramos al ventilador como un producto simple. Si la empresa además de los ventiladores, fabrica y vende los motores, en este caso consideramos al ventilador como producto compuesto.

En base a esta clasificación de los productos en simples y compuestos, el diagrama de clases anterior se modifica agregándole la herencia a Producto Simple y a Producto Compuesto, y la composición indicada. Esto da lugar al Diagrama de Clases 2.

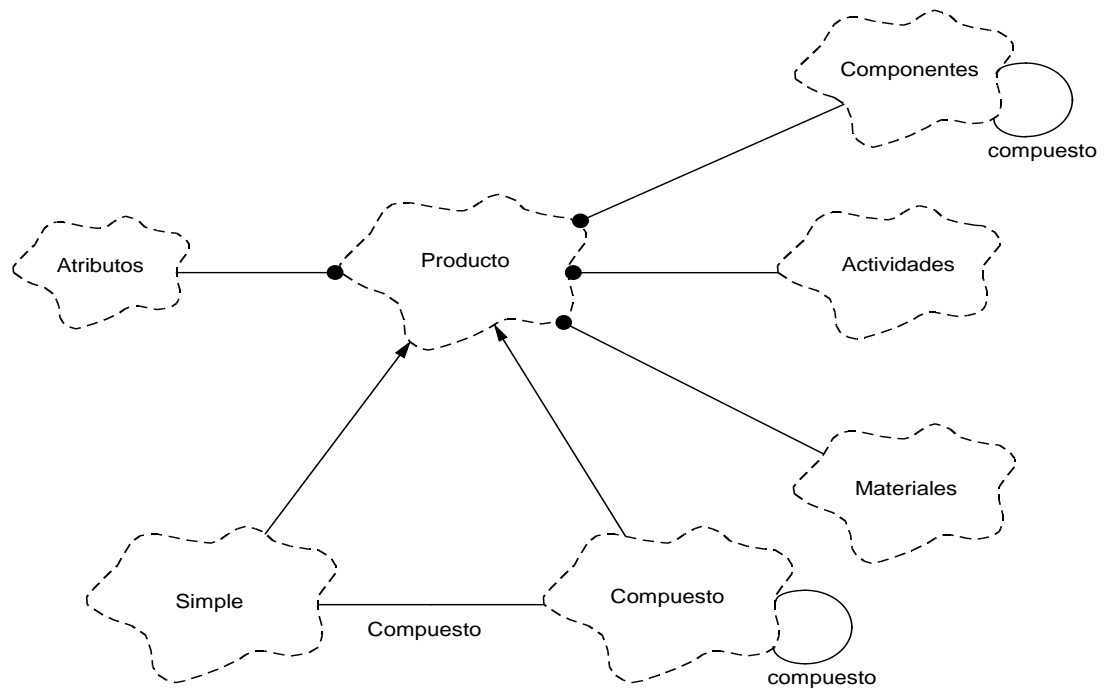


Diagrama de Clases 2

Para poder agrupar a los productos con características similares (Clases de productos) se incorpora la idea de *Familia*. Una familia contiene la referencia a todos sus miembros. En realidad la clase ya es la familia de productos, lo que le falta es la referencia a todas las instancias de esta clase. Ej: si definimos una mesa XX, esta es la familia de las mesas XX, cada instancia de mesa XX serán sus miembros. La instancia incorporará los valores específicos para los distintos atributos de la clase. No es buena medida poner en la clase la lista de objetos que “instanció”, ya que de esta manera cada objeto “instanciado” tendría la lista de objetos. Entonces definimos una nueva clase que será la contenedora de las listas de objetos “instanciados” por una clase X. Esto da origen al Diagrama de Clases 3.

Todas las clases así planteadas tienen una estructura (atributos) y comportamiento definido. Es decir todas las *componentes* tendrán los mismos atributos y los mismos mensajes, todas las actividades tendrán los mismos atributos y métodos, y así cada clase. Lo que variará en cada instancia de estas clases serán los valores particulares que se asignen a ésta, y serán propias de cada producto “instanciado”.

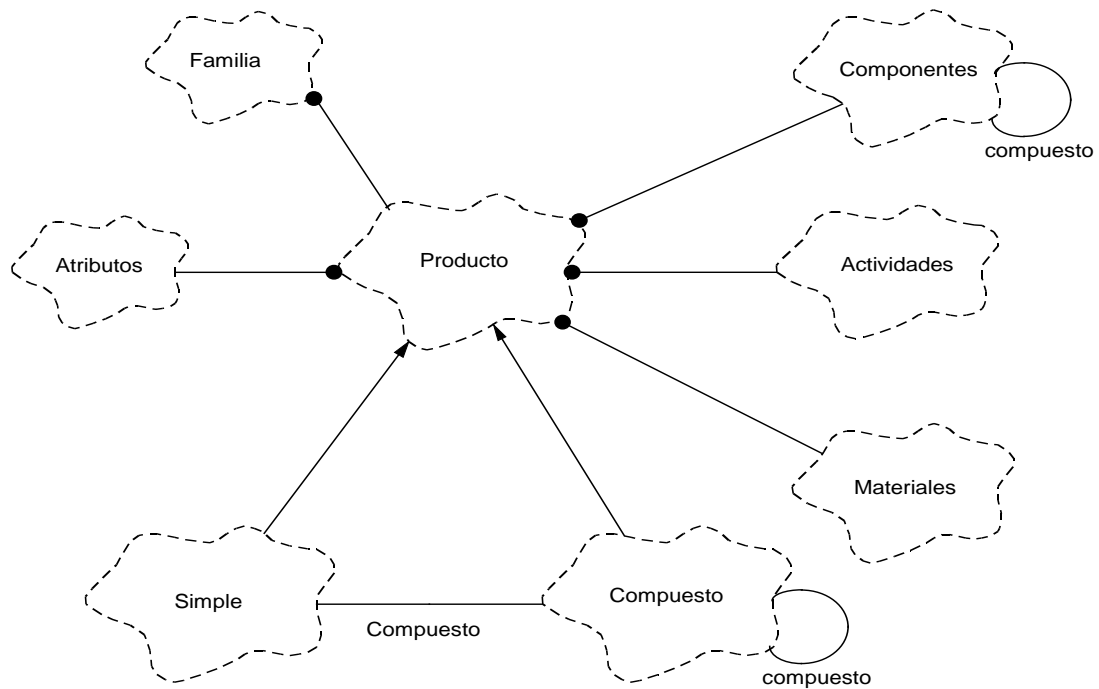


Diagrama de Clases 3

Los *materiales* y *componentes* requieren definir cuáles son sus medidas o valores a utilizar. Para poder trabajar con distintos tipos de unidades de medidas se define una clase *medidas* con diferente especialización. Estas también deben definir sus fórmulas de cálculo para poder aplicar a los valores concretos que se suministren. Representamos esto a través del Diagrama de Clases 4.

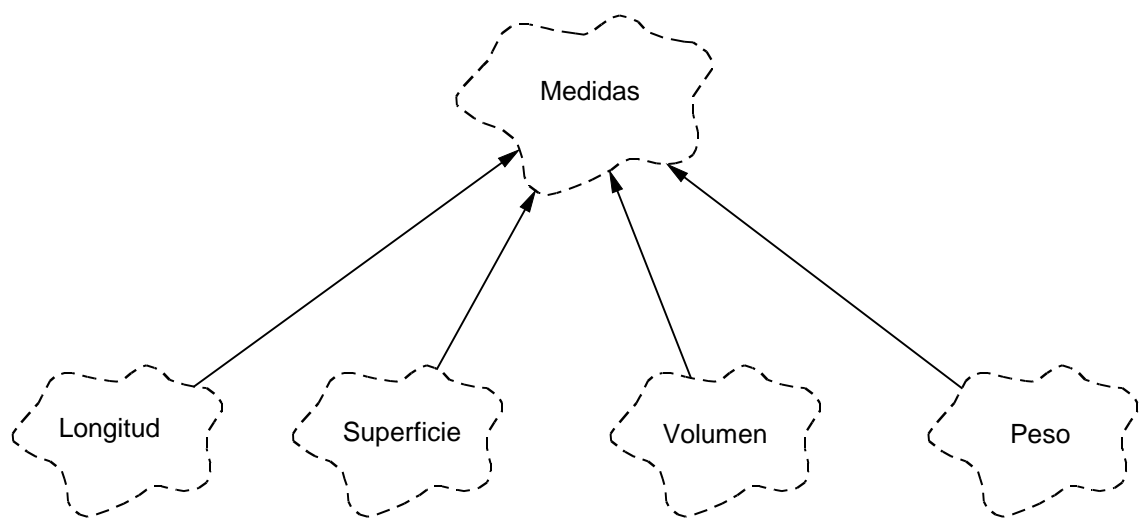


Diagrama de Clases 4

Las *componentes* y *materiales* deben hacer uso de esta clase para determinar la unidad de medida de los elementos que “instancien”. Por ejemplo: la tapa de una mesa (componente) debe decir que su medida es superficie, y su ancho y largo. La relación de uso se representa en el Diagrama de Clases 5.

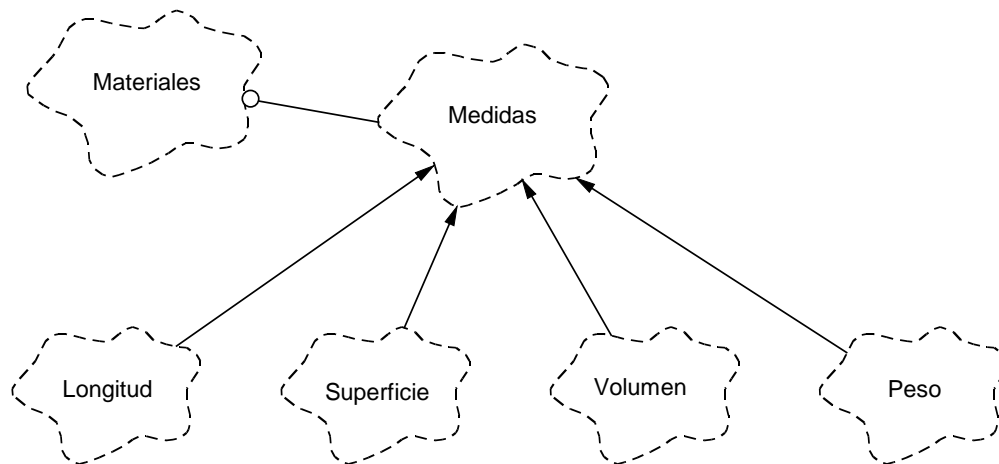


Diagrama de Clases 5

En cuanto al comportamiento de las clases definidas, para obtener flexibilidad en las definiciones, se establece que cada clase (excepto *Producto* y *familia*) debe definirse en forma individual, y luego utilizarse para conformar un producto en particular.

Esto plantea la situación simple donde cada clase tendrá sus actividades de *crear* una nueva instancia, *modificar* una instancia existente o *borrar* una instancia existente. En cambio para producto, cada una de las instancias de sus componentes que quiera utilizar ya deben existir para poder “instanciarlas”, esto implica que debe comunicarse con las clases de interés para ver si existe la instancia que el necesita. *Producto* también tendrá sus operaciones básicas de crear, modificar y borrar. *Producto* debe comunicarse con *familia* para avisar que hay un nuevo producto (o que fue eliminado).

Hemos presentado así una comprensión básica del problema, el paso siguiente fue realizar un análisis detallado de la estructura de cada clase y de sus responsabilidades (métodos). Para esto hemos desarrollado el *Diccionario de Clase* para cada una de las clases. A modo de ejemplo se muestra el diccionario de la clase *Producto*:

NOMBRE	<i>Producto</i>
DESCRIPCIÓN	<i>Producto que se diferencia del resto de los producidos por la empresa. Siempre pertenece a una familia determinada, no existiendo como ente aislado.</i>
PROPIEDADES	<i>nombre descripción identificador lista de componentes lista de actividades lista de materiales lista de atributos</i>
RESPONSABILIDADES	<i>crearse destruirse modificarse mostrar lista de componentes mostrar lista de atributos mostrar lista de materiales mostrar lista de actividades actualizar lista de miembros</i>

A partir de este análisis detallado se obtuvo información sobre las relaciones estructurales necesarias, (por ejemplo, que la clase producto necesitará de una lista de sus materiales, una lista de actividades y una lista de componentes), como así también un detalle de los atributos de cada clase, y un detalle de los métodos requeridos por la clase.

CONCLUSIONES

Como conclusión del presente trabajo debemos destacar en primer lugar que hemos adoptado una política de desarrollo evolutiva para el prototipo de generador del SSD por lo que tanto el diseño de la estrategia de manejo de la información, como el análisis y diseño del subsistema fue avanzado hasta un nivel que consideramos satisfactorio y luego pasamos a la etapa de programación (actualmente en desarrollo). Los resultados de esta primer versión académica del prototipo posiblemente sugieran cambio tanto en la estrategia de manipulación de la información como del diseño del sistema. Por esta razón el objetivo del presente trabajo no ha sido mostrar un resultado, sino la estrategia desarrollada, los pasos seguidos y los criterios de análisis utilizados.

En segundo lugar, cabe destacar que la descripción del subsistema se ha centrado exclusivamente en la información relativa al producto, esto es porque la información relativa a otros aspectos, que por cierto también interesan al momento de diseño es responsabilidad de otros subsistemas del SSD en desarrollo. Por ejemplo, la capacidad de producción, las características y disponibilidad de los materiales, las proyecciones de ventas, la política estratégica de la organización, los costos, etc., es información que soporta la actividad de diseño de producto pero que es responsabilidad de subsistemas distintos.

Finalmente queremos destacar que la implementación se está llevando a cabo utilizando el lenguaje de programación visual DELPHI, para lo cual debimos compatibilizar el diseño de objetos realizado siguiendo la metodología de Booch, y el lenguaje de programación. La más importante ha sido quizás la necesidad de almacenamiento de objetos en tablas relacionales.

REFERENCIAS

- Adamopoulos G., N. Karacapilidis & S. Pantazopoulos, *Production Management in the Textil Industry Using the "YFADI" Decision Support System*, Comp. & Chem. Engng Vol **18**, pp s577-s583, (1994).
- Booch, G., *Object Oriented Analysis and Design*, J.Wiley & Sons (1993).
- Erens, F. A. McKay & S. Bloor, *Product Modelling Using Multiple Levels of Abstraction Instances as Types*, Computers in Industry, 24, pp 17-28 (1994)
- Lorino, P. *El Control de Gestion Estratégico. La Gestión por Actividades*. Macombo, Barcelona, (1993).
- Niehaus, R.J., *Evolution of the Strategy and Structure of a Human Resource Planning DSS Application*. Decision Support Systems, 14, pp 187-204, (1995).
- Parker, S., E.M. Malstrom, L.M. Irwin & G. Ducote, *A Decision Support System for Personnel Scheduling in a Manufacture Enviroment*, Comp. Ind. Engineering, Vol. 27, 1-4, pp 185-188, (1994). Rico M., M. Rico, O. Yuschak, M.L. Taverna, J. C. Ramos, M.R Galli and O. Chiotti, *Decision Support Systems Generator for Industrial Companies*, Proceeding of 21th International Conference on Computers & Industrial Engineering. Puerto Rico, Marzo 1997.
- Sage, A.P., *Decision Support Systems Engineering*, John Wiley & Sons (1991).