

## DesProc 1.0.

### Una herramienta para descomposición formal funcional de procesos

Tatiana Arias Manrique  
Javier Enrique Mora Díaz  
Ricardo Llamosa-Villalba  
Grupo de Investigación en Ingeniería del Software  
Escuela de Ingeniería de Sistemas e Informática  
Universidad Industrial de Santander  
E-mail: rllamosa@uiscol.uis.edu.co  
Apartado Aéreo 678, Ciudad Universitaria UIS  
Bucaramanga, Colombia

#### RESÚMEN

“Descomponer Procesos”, es comúnmente difícil y complejo [2][3][5][6]; la cantidad de variables, el conocimiento profundo del discurso del problema en evaluación y la subjetividad de las especificaciones, respecto a la constitución de un proceso, desencadena innumerables alternativas de modelado. El estudio del álgebra de Adler, facilita y formaliza, la labor de descomposición de procesos a través del flujos de datos. Este instrumento, con reglas claras y concisas, obtiene una misma descomposición, eliminando la influencia del criterio personal<sup>1</sup>. Este trabajo facilita la investigación de la descomposición de procesos de diversa índole, haciendo énfasis en procesos de ingeniería del software. Este hecho nos llevó a fabricar una herramienta que apoya la descomposición de procesos. Esta “máquina” con resultados gráficos y algebraicos, permite verificar la eficiencia y eficacia<sup>2</sup> de los criterios expuestos por Adler. Las debilidades y fortalezas descubiertas en nuestro trabajo de investigación se mencionan en este documento. Particularmente, la obtención de sentencias minimales<sup>3</sup>.

#### 1. FUNDAMENTACION

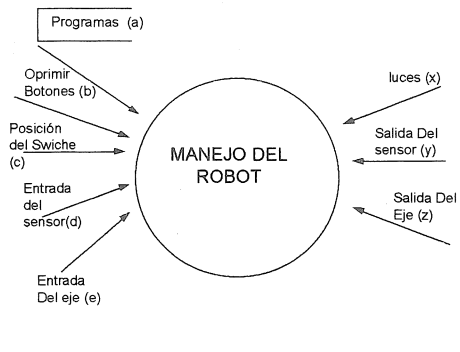
La descripción inicial de un proceso se da en un diagrama funcional denominado: "diagrama de contexto", que incluye variables de entrada, la función de contexto (caja negra) y las variables de salida. Realizar la descomposición, implica: 1) encontrar las relaciones existentes entre las variables de entrada

<sup>1</sup> Significa, desaparecer la posibilidad de hacer pruebas, para observar que descomposición resulta mas satisfactoria y plantear, así, el problema de otra forma, en vez de hacer diferentes descomposiciones para un mismo problema.

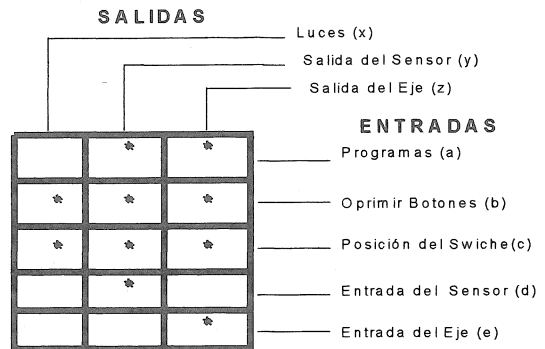
<sup>2</sup> Se pretende validar las reglas de Adler, en el proceso de interpretación, para todos los modelos que se propongan, es decir, si con ellas, en verdad, se pueden determinar, que elementos, son redundantes al final de dicho proceso. El control parte del estudio de casos con la asesoría de personas conocedoras de dichos casos.

<sup>3</sup> Una sentencia es minimal cuando una variable de salida se produce por uno y solo un proceso. Las sentencias no minimales no son contempladas en el álgebra de Adler.

y las variables de salida en una "matriz de entrada y salida": Las relaciones indican: que entradas son necesarias para la obtención de cada una de las salidas definidas<sup>4</sup>. 2) Definidas las entradas, las salidas y las relaciones, se aplican una serie de operadores algebraicos, propuestos por Mike Adler<sup>5</sup> [1]; Sustitución exacta, Sustitución de subconjunto, Sustitución débil, Absorción, Colección y Conexión, buscando obtener una sentencia minimal<sup>6</sup>. De no obtenerse una sentencia minimal, se lleva a cabo el proceso de eliminación de variables<sup>7</sup> <sup>8</sup>. 3) Obtenida la sentencia minimal se verifica la calidad de la descomposición; denominada "Interpretación". La interpretación consiste en aplicar un conjunto de reglas para obtener las sentencias iniciales dadas en la matriz de relaciones. Hecho que puede lograrse o no. La interpretación se hace sentencial o matricialmente. Un ejemplo sobre la funcionalidad de un robot, nos permite conceptualizar el modelo: La figura No. 1a, presenta el "Gráfico de Contexto" con las variables de entrada: *a, b, c, d, e* y las variables de salida: *x, y, z*. La figura No. 1b, describe las relaciones entre las variables de entrada y las variables de salida.



1a. Diagrama de Contexto



1b. Matriz de Entrada v Salida

Figura No. 1 Contexto Funcional de un proceso

Cada una de las relaciones definidas se denota de la siguiente manera: (*a*->*y,z*), (*b*->*x,y,z*), (*c* -> *x, y, z*), (*d* ->*y*), (*e* -> *z*); sentencias iniciales que corresponden a la representación algebraica de la matriz. Aplicar los operadores tiene el siguiente orden: 1) Sustitución exacta; 2) Colección; 3) Sustitución de

<sup>4</sup> Relación tipo "es utilizada para producir".

<sup>5</sup> Mike Adler, es consultor y diseñador de Software de procesos. Trabaja en el desarrollo de sistemas operativos y es miembro de la IEEE: Computer Society y de "The Association for Computing Machinery".

<sup>6</sup> Una sentencia no minimal refleja una descomposición, en la cual, se obtienen subprocesos diferentes con salidas idénticas, lo que indica que dicha descomposición, no es válida.

<sup>7</sup> Mike Adler no considera la posibilidad de llegar a sentencias no minimales al final del proceso de descomposición. Ante tal la ausencia de orientación en este sentido, los autores de este proyecto definieron pautas para la obtención de sentencias minimales cuando no obtienen directamente.

<sup>8</sup> Siempre y cuando, se esté de acuerdo en que las relaciones no afecten la minimalidad de la sentencia.

Subconjunto; 4) Conexión; 5) Absorción; 6) Substitución Débil. Los operadores se aplican a cada transformada o par de transformadas, en la lista de transformadas, para producir la sentencia minimal. En cada lista, de entrada y salida, de las nuevas sentencias, que vayan apareciendo, se les re-aplica los operadores. Después de cada minimización, la transformada o par de transformadas, resultantes, se reemplazan en la lista de transformadas por la sentencia minimizada, y los operadores son re-aplicados a la lista de transformadas para continuar con el proceso de minimización. Aplicando el proceso, tenemos:

Operación	Entrada	Sustitución	Resultado
1)Substitución Exacta	$(a \rightarrow y, z), (b \rightarrow x, y, z), (c \rightarrow x, y, z), (d \rightarrow y), (e \rightarrow z)$	$(b \rightarrow x, y, z) \langle \{x, y, z \leftarrow (c \rightarrow x, y, z)\}$	$(a \rightarrow y, z), (b \rightarrow (c \rightarrow x, y, z)), (d \rightarrow y), (e \rightarrow z)$
2)Colección	Resultado anterior	$(b \rightarrow (c \rightarrow x, y, z)) \langle \{b\}$	$(a \rightarrow y, z), (b, c \rightarrow x, y, z), (d \rightarrow y), (e \rightarrow z)$
3)Subconjunto	Resultado anterior	$(b, c \rightarrow x, y, z) \langle \{y, z \leftarrow (a \rightarrow y, z)\}$	$(b, c \rightarrow x, (a \rightarrow y, z)), (d \rightarrow y), (e \rightarrow z)$
4)Conexión	Resultado anterior	$(b, c \rightarrow x, (a \rightarrow y, z)) \langle \{(d \rightarrow y)\}$	$(b, c \rightarrow x, (a \rightarrow y, z), (d \rightarrow y)), (e \rightarrow z)$
5)Subconjunto	Resultado anterior	$(a \rightarrow y, z) \langle \{y \leftarrow (d \rightarrow y)\}$	$(b, c \rightarrow x, (a \rightarrow z, (d \rightarrow y))), (e \rightarrow z)$
6)Conexión	Resultado anterior	$(b, c \rightarrow x, (a \rightarrow z, (d \rightarrow y))) \langle \{(e \rightarrow z)\}$	$(b, c \rightarrow x, (e \rightarrow z), (a \rightarrow z, (d \rightarrow y)))$
7)Subconjunto	Resultado anterior	$(a \rightarrow z, (d \rightarrow y)) \langle \{z \leftarrow (e \rightarrow z)\}$	$(b, c \rightarrow x, (a \rightarrow (e \rightarrow z), (d \rightarrow y)))$

Terminada la descomposición, la sentencia minimal se Interpreta;

a)  $(b, c \rightarrow x, (a \rightarrow (e \rightarrow z), (d \rightarrow y)))$ . b)  $(a \rightarrow (e \rightarrow z), (d \rightarrow y))$ . c)  $(e \rightarrow z)$ . d)  $(d \rightarrow y)$

Ahora, determinamos, los nodos procesos:

Nodo Proceso	Nodo Proceso	Nodo Proceso	Nodo Proceso
a) $(b, c \rightarrow x)$	b) $(a \rightarrow \text{null})$	c) $(e \rightarrow z)$	d) $(d \rightarrow y)$

Los nodos proceso enlazados resultantes son:

Enlace	Enlace	Enlace	Enlace
a) $(b, c \rightarrow x) \Rightarrow (a \rightarrow \text{null})$	b) $((a \rightarrow \text{null}) \Rightarrow (e \rightarrow z), (d \rightarrow y))$	c) $(e \rightarrow z)$	d) $(d \rightarrow y)$

Esto especifica los siguientes flujos locales:

Desde el nodo Proceso	Al Nodo Proceso
1 $(b, c \rightarrow x)$	2 $(a \rightarrow \text{null})$
2 $(a \rightarrow \text{null})$	3 $(e \rightarrow z)$
2 $(a \rightarrow \text{null})$	4 $(d \rightarrow y)$

El gráfico de interpretación resultante es el DFD que se muestra en la Figura 2. El analista nombra los procesos y los flujos de datos locales para producir el DFD final de la descomposición.

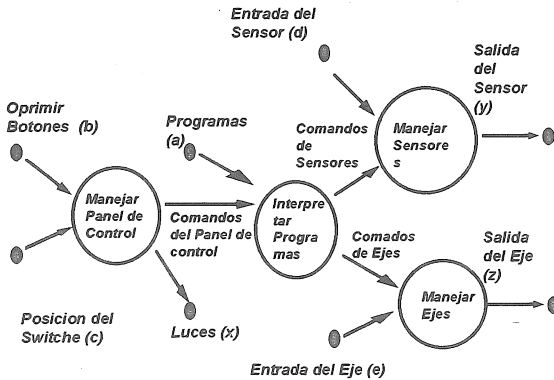


Figura No. 2 Grafo de Interpretación

Analizamos, ahora, la calidad de la descomposición, para verificar si aparecen nuevas relaciones; Esta calidad puede ser trivial, óptima o factible<sup>9</sup>. Una descomposición trivial es idéntica a la sentencia inicial e implica la no aplicación de operadores. Una descomposición óptima tiene una única solución y la descomposición factible tiene múltiples

soluciones; resulta cuando se debe usar alguna sustitución débil no conservativa o el operador de conexión (genera parejas que no necesariamente existían).

## 2. MODELO DE INGENIERIA

### 2.1. Recursos u Objetos.

En el diseño se ha considerado la necesidad de definir objetos que permitan el manejo de la información e intervengan en la descomposición de un proceso, tal como se ha explicado en la fundamentación. El propósito de cada uno de estos objetos se describe a continuación:

Objeto	Operaciones
Gráfica de Contexto	Definir, eliminar, añadir, ordenar, nombrar y actualizar las variables que intervienen en un proceso.
Matriz de Relaciones	Permitir visualizar, crear y modificar las relaciones que se consideren necesarias en la matriz de relaciones.
Matriz de Interpretación	Visualizar e identificar relaciones actuales y nuevas entre las variables de Entrada y Salida producto de la interpretación.
Sentencias iniciales	Visualizar sentencias algebraicas y las relaciones entre las variables de entrada y salida definidas en la matriz inicial.

<sup>9</sup> Una calidad es trivial si la relación es idéntica a la sentencia inicial o es un término. Una calidad es óptima si la relación es equivalente a la inicial y es no trivial. Una calidad es factible si la relación es no equivalente a la sentencia inicial y es no trivial.

Objeto	Operaciones
<b>Flujo de Datos</b>	Mostrar la descomposición obtenida en forma de diagrama de flujo de datos de tal manera que se puedan apreciar los flujos , las variables de entrada y de salida de cada uno de los nodos y el orden de precedencia de los mismos.
<b>Interpretación</b>	Visualizar la interpretación algebraica comparando las sentencias iniciales con las finales para establecer las relaciones adicionales generadas con la descomposición.
<b>Variables</b>	Identificar y Definir estructuras para representar diferentes conjuntos de variables de entrada y salida.
<b>Relaciones</b>	Identificar y mostrar la jerarquía de los procesos respecto a cada una de las relaciones definidas entre las variables de entrada y de salida.

## 2.2. Macroprocesos.

Describimos, a continuación, cada Macroproceso (figura No. 3) en los que se encuentran cada uno de los procesos diseñados para DesProc 1.0.

Administración de Archivos: Operaciones para crear, recuperar, guardar, renombrar o ubicar archivos y operaciones para la selección de equipos de impresión e impresión del contenido de la ventana activa.

Edición: Operaciones de edición y copia de cualquier presentación, eliminación y selección o inhibición de variables de presentación del "gráfico de contexto".

Visualizar presentaciones: Operaciones para visualizar gráficos de contexto, matriz de relaciones, sentencias iniciales, descomposición algebraica, DFD e interpretación de sentencias y matrices.

Creación de Variables: Operaciones para insertar variables de entrada y variables de salida en el gráfico de contexto.

Manejo de Ventanas: Operaciones para la organización y gerencia des ventanas.

## 3. PROTOTIPO

A partir del diseño concebido y descrito en 2, se desarrolló la herramienta en Borland Pascal (ver.7.0) para Windows. El prototipo concretó los procesos de gráfico de contexto, matriz, ventana de sentencias minimales, descomposición en sentencias, procesos y DFD's, interpretación sentencial, interpretación y representación matriz y ayudas.

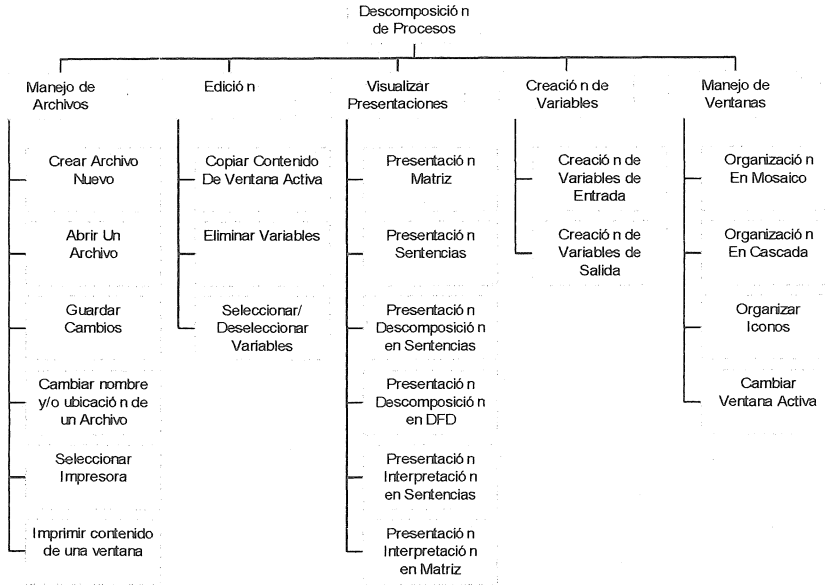


Figura No. 3 : Macro Procesos de DesProc 1.0

**3.1. Gráfico de Contexto.**

Con este objeto se inicia la aplicación DesProc1.0; definiendo o insertando las variables de entrada y de salida, colocando nombres a las mismas o dejando los sugeridos por la herramienta (Ent1, Ent2...para las variables de entrada y Sal1, Sal2,...para las variables de Salida) o cambiando el orden de las variables de entrada.

**3.2 Matriz.**

EL requisito de operación de este objeto es la existencia del "Gráfico de contexto"; a través del menú "Ver" y el comando "Matriz" se podrá obtener la ventana Matriz. El objeto matriz define las relaciones existentes entre las variables de entrada y una o más variables de salida. Cuando el objeto se activa aparece una matriz de *n* filas, que representan las variables de entrada, y *m* columnas, que representan las variables de salida. En los encabezados de cada una de las filas y columnas aparecerán seis letras iniciales de cada variable. Para establecer las relaciones entre las variables de entrada y salida, solo es necesario hacer un clic<sup>10</sup> en el 'Mouse', en la casilla de intersección entre la entradas y las salidas que se

<sup>10</sup> El proceso es igual cuando se desee desactivar alguna relación que se haya colocado con anterioridad.

deseen relacionar. Este objeto (ventana) una vez activo, no podrá cerrarse, porque es fundamental en la descomposición.

### **3.3. Ventana Sentencias Iniciales.**

Definidas las variables que intervienen en el proceso y las relaciones entre las mismas, es posible visualizar las relaciones en forma de sentencias. Es indispensable que todas las variables de entrada se encuentren relacionadas al menos con una variable de salida. Este objeto, activo, tiene la función de representar algebraicamente las relaciones en forma interna o externa (vista para el usuario). Esta ventana u objeto, a diferencia de Matriz y Gráfico de Contexto, puede cerrarse o activarse en el momento en que se requiera; ella refleja cualquier cambio realizado en "Gráfico de Contexto" y "Matriz".

### **3.4. Descomposición en Sentencias.**

El objeto que lleva a cabo el proceso de descomposición requiere de la existencia de matriz y grafo de contexto. El resultado de dicho proceso visualiza el producto en dos formas: sentencias (representación algebraica de la descomposición) y grafos de Diagramas de Flujo de Datos -DFD- (Representación de usuario, que muestra las variables con los seis primeros caracteres de los nombres asignados por el objeto "Gráfico de Contexto").

### **3.5. Descomposición en DFD.**

Este objeto muestra la otra forma de representar la descomposición a través de los DFD's. Su representación gráfica utiliza las siguientes convenciones: Los círculos azules representan cada uno de los subprocesos en los cuales se dividió el proceso inicial. Las flecha representan flujos de datos entre procesos. Las letras expresan la representación interna de las variables.

### **3.6. Interpretación Sentencias.**

Al igual que la descomposición, la Interpretación puede representarse de dos formas; sentencias o matrices. Las Sentencias son la representación algebraica de las relaciones resultantes de la descomposición del proceso; tienen una representación interna y representación externa. La representación interna muestra las variables de entrada y de salida con las letras asociadas según lo haya hecho el objeto "Gráfico de Contexto". La representación externa muestra las variables con los seis primeros caracteres de los nombres asignados en el "Gráfico de Contexto". Este objeto puede abrirse o cerrarse en cualquier momento.

### 3.7. Interpretación y Representación Matriz.

Este objeto presenta una Matriz con las mismas características de "Matriz". Las relaciones originales se representan con los círculos de color azul, mientras que las relaciones adicionales se encuentran representadas por medio de los círculos de color rojo. El objeto puede desactivarse en el momento que se requiera y activarse siempre y cuando, todas las variables de entrada tengan alguna relación con por lo menos una variable de salida.

### 3.8. Ayudas.

El objeto ayudas puede invocarse con el comando "Contenido" del menú "Ayuda" o a través de su icono correspondiente en la barra de herramientas. La ventana principal de ayudas, presenta los temas teóricos principales de la fundamentación y de la herramienta en sí misma. Se maneja a través de hipertextos o recuadros explicativos o a través de temas claves específicos.

## 4. PRUEBAS

Para la validación de la herramienta Software desarrollada se realizó un generador automático de pruebas y cerca de 10 pruebas funcionales con problemas de procesos reales. Para el caso de las pruebas funcionales presentamos un ejemplo (Tabla No. 1):

Tabla No. 1. Prueba funcionamiento No. 6. "Producción de Bienes Industriales".

<b>Evaluador</b>	Jaime Camacho
<b>Carrera</b>	Ingeniería Industrial
<b>Facultad</b>	Físico Mecánicas
<b>Tema o Proceso a Analizar</b>	Producción de Bienes Industriales

Descripción del proceso: En la producción de bienes industriales[4] se deben elaborar productos terminados con determinados estándares de calidad, tiempo y especificaciones técnicas. Las materias primas, el tipo de tecnología (maquinaria - equipo - procesos), la calificación del personal que interviene en el proceso, la gestión administrativa y el marco normativo y legal en que opera la empresa, son variables que tienen incidencia en el producto final.

Descripción de la descomposición: Este problema consta de siete variables de entrada y cuatro variables de salida. Las relaciones definidas entre las variables de entrada y de salida pueden verse en las siguientes figuras.

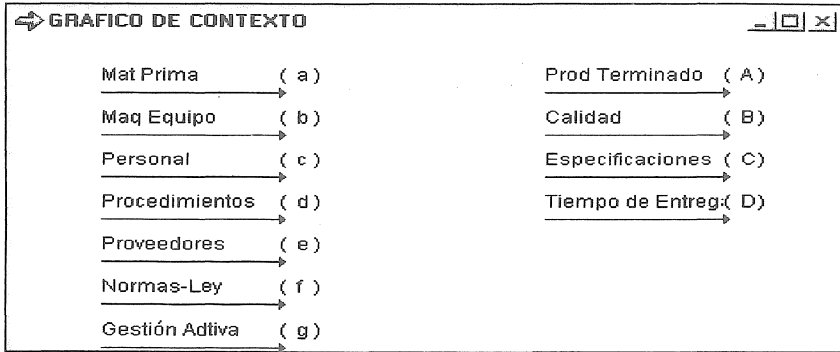


Figura 5. Gráfico de Contexto. "Producción de Bienes Industriales"

Figura 6. Matriz. Producción de bienes Industriales

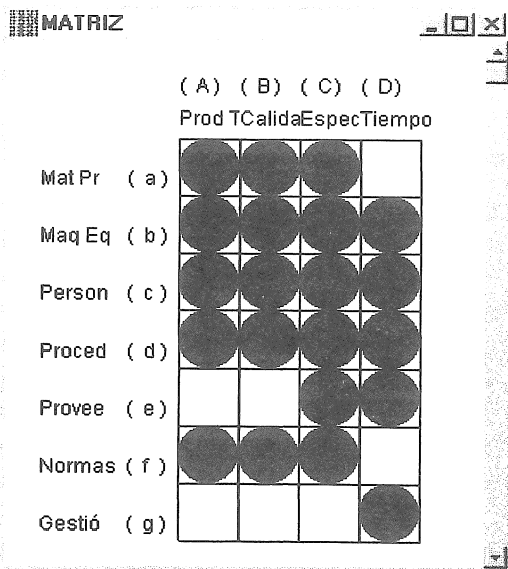


Figura 7. DFD. Producción de bienes Industriales

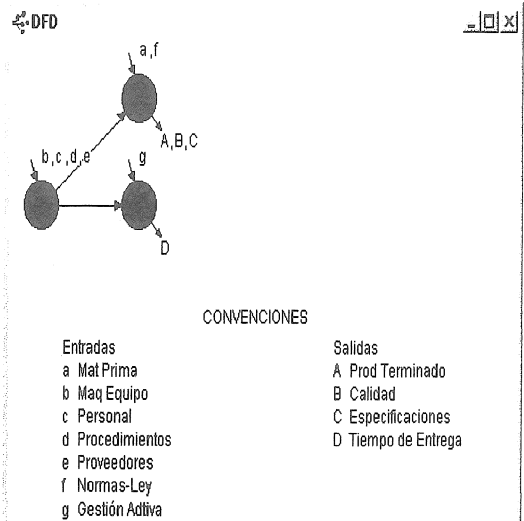
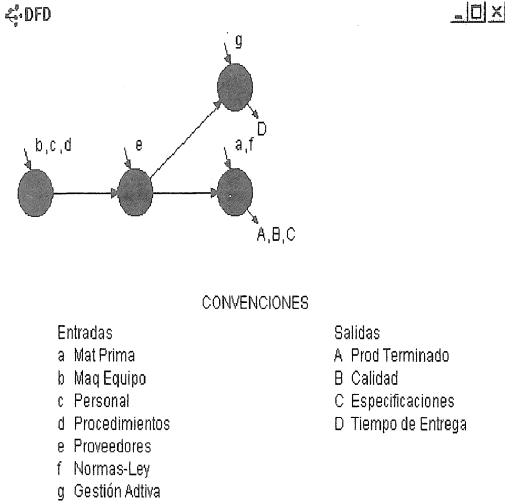


Figura 8. DFD. Producción de Bienes Industriales



El resultado de la descomposición es una sentencia minimal; la calidad de la descomposición medida en la matriz de Interpretación se cataloga como factible. Al igual que en algunos casos anteriores, aparecen relaciones que no se tenían en cuenta en el problema original como lo son (Proveedor -> Producto terminado) y (Proveedor -> Calidad). Si el analista considera que cualquiera de las dos relaciones se puede dar, se conserva la minimalidad y la factibilidad de la misma. Ninguna de las dos relaciones afecta el DFD o la estructura de los nodos subprocessos. Si se consideran las dos relaciones al mismo tiempo, la estructura del DFD

cambia, disminuyendo la cantidad de nodos subprocesso. Esto último conserva la minimalidad, y la calidad de la misma pasará de factible a óptima. Estos cambios se hacen siempre y cuando el analista los considere necesarios, teniendo en cuenta que estas relaciones no cambian la esencia del problema original. Según el DFD del problema original, los factores como maquinaria y equipo, personal, proveedores, procedimientos y normas de ley, afectan de manera indirecta el tiempo de entrega; el factor primordial para que un producto se entregue a tiempo es la gestión administrativa. Igualmente ocurre con la materia prima; posee determinadas especificaciones y es el factor más importante para que un producto terminado sea de óptima calidad.

5. CONCLUSIONES

5.1. A Nivel de Usuario.

- La rapidez con que se obtienen los resultados y la posibilidad de verlos simultáneamente en diferentes formas, facilita el proceso de prueba de diferentes definiciones para un mismo problema.
- Los usuarios que deseen profundizar sobre la forma en que opera el álgebra, encontrarán en la herramienta un excelente apoyo para indagar sobre la posibilidad obtener mejoras a la especificación teórica utilizada.

### 5.2. A Nivel de Desarrollo.

- La formalización proveída para el proceso de descomposición, elimina la subjetividad en la especificación de procesos; probar el hecho es ahora posible, obteniéndose descomposiciones “lógicas” para que los expertos consultados, en los casos analizados, los evalúen.
- La definición de un problema de flujos de datos mediante relaciones entre variables, sin considerar el grado de incidencia de unas sobre otras, sino la simple certeza de la influencia entre ellas, permite descubrir flujos que habían sido considerados de poca importancia. En algunos casos se llegan a descubrir incidencias desconocidas de una variable sobre otra.
- Los operadores del álgebra de Adler, en la casi totalidad de los casos, pueden aplicarse en un orden fijo; la aplicación sucesiva de un mismo operador sobre un lista de transformadas hace que éstas lleguen a un estado en el que ya no es posible aplicar dicho operador, y por lo tanto la única forma de continuar con el proceso de descomposición, es la prueba con otros operadores.
- El hecho de aplicar los operadores del álgebra de Adler, en diferente orden, conlleva<sup>11</sup> a la obtención de descomposiciones muy distintas.
- La medida de la calidad o interpretación de la descomposición es indicio de relaciones adicionales con respecto a las relaciones originales.
- Puesto que DesProc 1.0, formaliza el proceso de descomposición, los resultados serán, mas o menos lógicos, en la medida que sean mas acertadas las relaciones definidas entre las variables de entrada y las variables de salida; El hecho supone un buen conocimiento de las razones por las cuales se incluye cada variable de entrada o cada variable de salida en el proceso en estudio.

### 5.3. A Nivel de Aporte.

- La herramienta obtenida permite analizar la descomposición de cualquier proceso independientemente de su naturaleza; no tiene en cuenta las características físicas u abstractas representadas por cada variable.
- Dado que el álgebra de Adler no considera la posibilidad de llegar a sentencias “no minimales” no provee mecanismos para solucionar este problema; razón necesaria definir algunas políticas adicionales alternas para suplir esta deficiencia.

---

<sup>11</sup> Aunque en la mayoría de los casos las diferencias son mínimas

- DesProc 1.0, se concibe como eslabón o apoyo en la cadena de aplicaciones que puedan beneficiarse durante el proceso de especificación de productos software.
- La incidencia del orden de las variables de entrada en el resultado de la descomposición obtenida permitió desarrollar un mecanismo de análisis para interpretar dicha incidencia.
- El estudio del álgebra nos llevó a cuestionar el funcionamiento de la misma, haciendo algunas conjeturas acerca de los operadores; la aparición de un solo subproceso como el inicial o la inexistencia de ciclos en la descomposición.

### 6. Bibliografía

- [1] Addler M. *"An Algebra for Data Flow Diagram Process Decomposition"*. Transaction on Software Engineering. Vol. 14 #2 February de 1988.
- [2] Llamosa-Villalba R. "Specifications Systems Information- (SSI). Specifications and Requirements for Systems Information Model", XIV IASTED International Conference, Applied Informatics, Innsbruck, Austria. February 20-22, 1996.
- [3] Llamosa-Villalba R., *Evaluation and Synthesis of Computational Methods, Formalisms and Tools in order to develop Information Systems Specification*, System and Computer Science Engineering School, UIS, Bucaramanga, Colombia, February, 1996
- [4] Harrinton H.J. *Mejoramiento de los procesos de la empresa*, Santafé de Bogotá, MacGraw Hill, 1992, 96p.
- [5] Protecko et all, *"Towards the automatic generator of Software Diagrams"*, *IEEE Transactions on Software Engineering*, Vol. 17 No. 1, January 1991, 10-21p..
- [6] Arndt T. and Gersoy Angela, *"The Composition of data flow"*, Proc. 4 4, int Conf. Software eng. Knowledge Eng. Public by *IEEE Computer Society, Los Alamitos, CA, USA*, 1992, 560-566p.