

36
20js

**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**



FACULTAD DE INGENIERIA

**SISTEMA TRADUCTOR DE
INTERCAMBIO ELECTRONICO DE DATOS**

T E S I S

**QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION
P R E S E N T A N :
RICARDO GABRIEL GONZALEZ GARCIA
JESUS MIGUEL PASTRAN RODRIGUEZ
LETICIA REYES TORRES
SUSANA YAÑEZ RAYON**

DIRECTOR DE TESIS

M.I. LAURO SANTIAGO CRUZ



MAYO 1992

**TESIS CON
FALLA DE ORIGEN**



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Tabla de Contenido

CAPITULO 1

introducción

| | |
|---|----|
| 1.1 ANTECEDENTES | 1 |
| 1.1.1 Una perspectiva historica. | 3 |
| 1.1.2 Estándares EDI | 3 |
| 1.1.3 Elementos y flujo de datos en un sistema EDI | 4 |
| Computadora que genera y recibe el mensaje | 4 |
| Modem | 5 |
| Red de comunicaciones pública o privada | 5 |
| Base de datos fuente y destino | 5 |
| Software de traducción EDI | 5 |
| Software de Comunicaciones | 5 |
| Unidades de Información | 6 |
| Elementos de Control | 6 |
| Sesión de Comunicaciones EDI | 7 |
| 1.1.4 Maneras de reaccionar en EDI, formas reactiva y promotora | 7 |
| 1.1.5 Madurez de la tecnología EDI | 9 |
| 1.1.6 El lugar de EDI en la estructura de negocios | 11 |
| 1.1.7 EDI y sus Interfaces con aplicaciones existentes | 11 |
| 1.1.8 Direcciones futuras | 12 |
| 1.1.9 Puntos legales en EDI | 12 |
| 1.2 PROBLEMATICA | 12 |
| 1.3 OBJETIVO Y ALCANCES | 13 |

CAPITULO 2

análisis del sistema

| | |
|---|----|
| 2.1 ANALISIS DEL SISTEMA | 15 |
| 2.2 ALCANCES | 15 |
| 2.3 CARACTERISTICAS Y BENEFICIOS | 15 |
| 2.4 USUARIOS | 16 |
| 2.5 DEFINICIONES Y TERMINOS ESPECIALES | 16 |
| 2.6 ELEMENTOS DE UN SISTEMA EDI | 17 |
| 2.7 REQUERIMIENTOS | 19 |
| 2.8 DIAGRAMA CONCEPTUAL DE USUARIO | 20 |
| 2.8.1 Sistema Traductor de Formatos EDI | 21 |

Sistema Traductor de Intercambio Electrónico de Datos

| | | |
|-------|--|----|
| 2.9 | DIAGRAMA DE FLUJO DE DATOS. | 26 |
| 2.9.1 | Descripción de Diagramas de Flujo de Datos. | 27 |
| 2.10 | DESCRIPCIÓN DE PROCESOS | 33 |
| 2.11 | DESCRIPCIÓN DE ENTRADAS /SALIDAS | 38 |
| 2.12 | DIAGRAMA DE ESTRUCTURA DE LA INFORMACIÓN (Diagrama lógico de la base de datos) | 41 |
| 2.13 | DICCIONARIO DE DATOS | 43 |

CAPITULO 3

diseño del sistema

| | | |
|---------|---|----|
| 3.1 | AMBITO | 45 |
| 3.1.1 | Objetivos del sistema. | 45 |
| 3.1.2 | Hardware, software e Interfaces humanas. | 45 |
| 3.1.3 | Principales funciones del software. | 46 |
| 3.1.4 | Base de datos definida externamente. | 47 |
| 3.1.5 | Principales ligaduras y limitaciones del diseño. | 47 |
| 3.2 | DOCUMENTOS DE REFERENCIA. | 47 |
| 3.2.1 | Descripción de software existente | 47 |
| 3.2.2 | Descripción de sistemas existentes | 48 |
| 3.3 | DESCRIPCION DEL DISEÑO. | 50 |
| 3.3.1 | Metodología de diseño | 50 |
| 3.3.1.1 | Acoplamiento y Cohesión | 51 |
| 3.3.1.2 | Revisión del flujo de datos. | 52 |
| 3.3.2 | Estructura del programa | 57 |
| 3.3.2.1 | Descripción de las estructuras del programa | 57 |
| 3.3.2.2 | Descripción de las heurísticas de diseño | 58 |
| 3.3.2.3 | Descripción de las estructuras de programa finales. | 59 |
| 3.3.3 | Carta de estructura de software | 63 |
| 3.4 | DESCRIPCION DE MODULOS | 67 |
| 3.5 | ESTRUCTURA DE ARCHIVOS Y DATOS GLOBALES | 68 |
| 3.6 | PROVISIONES DE PRUEBA. | 69 |

CAPITULO 4

desarrollo

| | | |
|-------|--|----|
| 4.1 | CRITERIOS PARA SELECCIONAR EL LENGUAJE Y LAS HERRAMIENTAS DE DESARROLLO. | 71 |
| 4.1.1 | El proceso de traducción. | 71 |
| 4.1.2 | Características de los lenguajes de programación. | 71 |
| 4.1.3 | Un Modelo Sintáctico/Semántico | 71 |
| 4.1.4 | Una visión de Ingeniería. | 71 |
| 4.1.5 | Elección de un lenguaje. | 73 |

Tabla de Contenido

| | |
|--|----|
| 4.1.6 Lenguajes de programación e Ingeniería de software. | 73 |
| 4.1.7 El estilo de codificación. | 74 |
| 4.1.7.1 Documentación del código. | 74 |
| 4.1.7.2 Declaración de datos. | 75 |
| 4.1.7.3 Construcción de sentencias. | 75 |
| 4.1.7.4 Entrada/Salida | 75 |
| 4.1.8 Eficiencia | 76 |
| 4.1.8.1 Eficiencia en código | 76 |
| 4.1.8.2 Eficiencia en memoria | 77 |
| 4.1.8.3 Eficiencia en la entrada/salida | 77 |
| 4.1.9 Consideraciones sobre la elección del lenguaje de programación y el equipo utilizado. | 78 |
| 4.2 PRUEBAS | 79 |
| 4.2.1 Objetivos de la prueba | 79 |
| 4.2.2 Flujo de información de la prueba | 79 |
| 4.2.3 Diseño de casos de prueba | 80 |
| 4.3 PRUEBAS REALIZADAS SOBRE EL SISTEMA. | 81 |

CAPITULO 5 conclusiones

83

GLOSARIO

APENDICE A

APENDICE B

BIBLIOGRAFIA

INDICE

C

CAPITULO 1

Introducción

1.1 ANTECEDENTES

En los años 60's las compañías tenían requerimientos internos de procesamiento de datos, en los cuales sólo se introducían los datos a las computadoras para generación de reportes.

En los 70's cambió el esquema, ya que se deseaba compartir información de manera interna, es decir entre los departamentos de una misma compañía.

En la década de los 80's se generó la necesidad de compartir información de manera externa, es decir, entre compañías.

En el lapso transcurrido entre 1960 y 1980, se utilizaron diversos medios para transmitir la información entre empresas, destacándose el correo, el teléfono, recientemente el FAX, y los servicios de mensajería (figura 1.1).

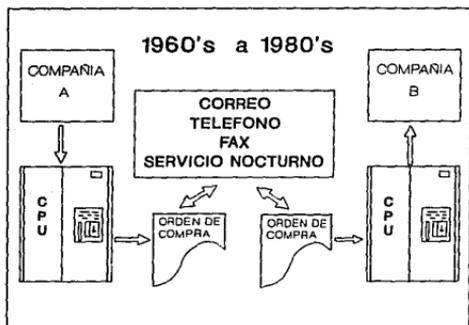


Figura 1.1. Medios de transmisión de información.

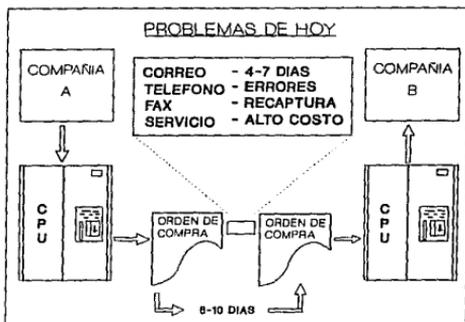


Figura 1.2. Problemas con medios de transmisión actuales.

Los problemas principales que se tienen con estos medios de comunicación son los siguientes (figura 1.2):

- Correo : el período de entrega de la información es de 4 a 7 días
- Teléfono : presenta muchos errores
- FAX : hay que recapturar la información
- Mensajería : sus costos son altos

La solución a estos problemas la proporciona el Intercambio Electrónico de Datos (EDI: Electronic Data Interchange), cuyo objetivo es el de transmitir datos de manera electrónica que puedan ser procesados por una computadora, utilizando formatos estándar.

La definición de los estándares EDI es:

Un conjunto de reglas, acordadas, aceptadas, y adquiridas voluntariamente, mediante las cuales los datos son estructurados en formatos de mensaje para intercambio de información operativa o de negocios.

Sistema Traductor de Intercambio Electrónico de Datos

Al transmitir los datos utilizando estos estándares se obtienen grandes ventajas (figura 1.3), y se enumeran a continuación:

- Es un medio rápido
- Se generan menos errores
- No hay necesidad de recapturar la información
- Utiliza una vía de comunicación barata
- Menor costo de intercambio cuando la alternativa de tiempo real es utilizada.
- La flexibilidad de utilizar campos de longitud variable.
- Flexibilidad en su utilización - Puede ser utilizado para intercambio en tiempo real o en intercambio de información fuera de línea por cinta o disco flexible.
- El intercambio de información utilizando una computadora personal es posible cuando se usa la alternativa de tiempo real.
- Permite que más información sea capturada por el receptor de los datos sin costos adicionales de captura.

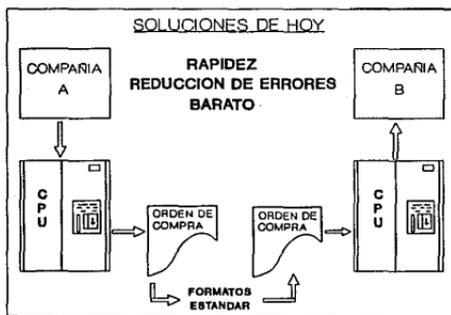


Figura 1.3. Ventajas de EDI

En la década de los 60's existían estándares específicos desarrollados por cada compañía y para su aplicación en particular, pero en los 70's surgieron diversas asociaciones con el fin de generar formatos especiales para algunas ramas industriales, como fue el caso de :

| ASOCIACION | APLICACION |
|--|-----------------|
| TDC (Transportation Data Coordinating Committee) | Transportación |
| UCS (Uniform Communication Standard) | Supermercados |
| WINS (Warehouse Information Network Standard) | Almacenes |
| ODETTE (Organization for Data Exchange Through Teletransmission In Europe) | Carros europeos |
| SWIFT (Society for Worldwide Interbank Financial Telecommunication) | Bancos |

En la década de los 80's se generaron los primeros estándares que cubrían a todas las industrias; además, se generaron subconjuntos de los que ya existían, para aplicaciones de ramas industriales, por ejemplo:

| ASOCIACION | APLICACION |
|--|----------------------|
| ANSI (American National Standards Institute) X.12 | Todas las industrias |
| CIDX (Chemical Industry Data eXchange) | Química |
| AIAG (Automotive Industry Action Group) | Automóviles |
| VICS (Voluntary Interindustry Communication Standards) | Ventas al por menor |
| TALC (Textil / Apparel Linkage Council) | Textiles |

Para los años 90's se planean generar estándares públicos y genéricos internacionales, que serán llamados EDIFACT.

1.1.1 Una perspectiva histórica.

El primer estándar EDI fué publicado hace solo trece años.

En una investigación realizada en 1989 por la Fundación Butler Cox, más del 43% de los miembros de las 300 compañías más grandes y avanzadas en Europa tenían aplicaciones mayores en EDI y sólo el 4% no tenía ninguna actividad con estos estándares.

En la investigación del estándar EDI se incluyeron todos los tipos de Intercambio electrónico de Información de negocios, para reemplazar las transacciones basadas en manejo de papel.

Actualmente en Estados Unidos de América se estiman entre 5,000 y 6,000 organizaciones gubernamentales y privadas que participan en el uso de estos formatos.

Históricamente, los siguientes eventos fueron significativos en la formación de EDI:

- 1). El desarrollo y publicación del primer conjunto de estándares (1974 - 1975).
- 2). La definición de la tecnología para el proceso de transmisión EDI (1976 - 1977).
- 3). La participación de ANSI y el desarrollo del standard X.12 (1978 - 1988).
- 4). La formación del Comité para la revisión del diccionario de datos EDI (1985 - 1988).
- 5). La coordinación Intercontinental de los estándares EDI (1985 - 1988).

1.1.2 Estándares EDI

Para promover la creación de estándares para la intercomunicación electrónica se formaron comités generales y particulares. Los particulares eran pequeños pero con fines específicos, tal fué el caso del TDCC para la transportación. Los comités generales fueron llamados fuerzas de tarea (*Task Forces*) y su función era mejorar el conjunto de códigos ya estandarizados, los pequeños comités llamados "grupos de trabajo" tenían como función diseñar la estructura de la Información para la transmisión de datos.

El procedimiento para el desarrollo de nuevos estándares estuvo integrado por los siguientes aspectos :

- 1). Formulación de invitaciones para participar en los grupos de trabajo.
- 2). Realización de reuniones entre las fuerzas de tarea y los grupos de trabajo.
- 3). Desarrollo de un conjunto de estándares para Intercambio Electrónico de Datos entre compañías, que permitieran modificaciones organizadas, de acuerdo a las necesidades del usuario. Además, el desarrollo debería ser con estructura de arriba hacia abajo (*top - down*), iniciando con amplios requerimientos (p. ej. una orden de compra) y finalizando con elementos de datos (p. ej. un nombre de cliente).
- 4). Publicación de los primeros estándares integrados para las Industrias de la transportación : aérea, automotriz, naviera y ferroviaria.
- 5). Posibilidad de modificaciones a los estándares creados, de acuerdo a las necesidades de los usuarios.
- 6). Las clases de comunicación definidas fueron :
 - computadora a computadora.
 - computadora a terminal.
 - terminal a computadora.
 - terminal a terminal.

Sistema Traductor de Intercambio Electrónico de Datos

La comunidad que utiliza los estándares EDI está agrupada en asociaciones:

- Internacionales UNECE (*United Nations Economic Council for Europe*) con estándares ISO (*International Standards Organization*)
- Nacionales en E.U. (ANSI estándares X.12) o de Industrias (por ejemplo TDCC).

El proceso que se sigue para el desarrollo y aprobación de un nuevo formato estándar en los Estados Unidos o de un cambio sobre los existentes, es el siguiente:

a.- Los usuarios de los estándares X.12 solicitan un nuevo conjunto de transacciones o una modificación;

b.- este requerimiento pasa a un subcomité de apreciación técnica, quien a su vez lo turna a los subcomités encargados del desarrollo de los estándares;

c.- después, el desarrollo es revisado por el comité de estándares X.12 para su aprobación;

d.- una vez aprobados, se les asigna el tipo de códigos para cada estándar, esto lo realiza el comité EDI de cada industria;

e.- el siguiente paso es insertar en las bases de datos de los estándares los nuevos formatos; y

f.- por último se procede a su publicación en los manuales ANSI X.12 y en las revistas relacionadas con EDI.

1.1.3 Elementos y flujo de datos en un sistema EDI

Los estándares, con una sintaxis definida y regidos por un conjunto de reglas, hacen posible combinar datos de requerimientos de comunicación para algunas industrias bajo un formato común. Los formatos de datos estandarizados permiten comunicar compañías entre sí,

computadora a computadora. El medio ambiente de estos estándares también provee estructura, generalidad, elementos de bases de datos y procedimientos para el desarrollo de nuevos formatos estándar.

Los documentos que se pueden intercambiar utilizando EDI van desde una orden de compra hasta una factura, o un aviso de embarque. Estos documentos se transmiten utilizando un conjunto de transacciones elementales como son: nombres, direcciones, descripciones, precios, cantidades, etc.

Las diversas combinaciones de estas transacciones primarias generan los diferentes documentos para intercambio.

Los elementos de un sistema EDI se dividen en dos tipos:

FISICOS

Hardware

Computadora que genera y recibe el mensaje
Modem: modulador - demodulador
Red de comunicaciones (pública o privada)

Software

Base de Datos fuente
Software de traducción
Software de comunicación
Base de Datos destino

LOGICOS

Unidades de información
Elementos de control
Sesión de comunicaciones

A continuación se explica brevemente cada uno de los elementos.

Computadora que genera y recibe el mensaje

En EDI no importa el tipo de computadora que se utilice. Se puede trabajar con Mainframes, Mini y Micro-computadoras de manera indistinta, debido a que el manejo de los estándares EDI es independiente del equipo en el cual se generen los mensajes.

Introducción

Modem

Modulador-Demodulador. Es un dispositivo que convierte la señal de una computadora en una serie de tonos para transmisión a través de una línea de comunicación. El proceso es revertido en el otro extremo de la línea.

Red de comunicaciones pública o privada

Las redes tienen como finalidad transferir e intercambiar datos entre computadoras. Son un medio de transporte para señales electrónicas que puede tener nodos conectados de tres formas: punto a punto, multipunto y *dial up*. Existen dos tipos, las que son controladas por el usuario (red privada) y las administradas por un tercero, que proporciona el servicio a varios usuarios (red pública), también llamadas VAN (Value Added Network).

Base de datos fuente y destino

La columna vertebral de EDI está en la utilización de bases de datos por lo que es presumible la revisión de las estructuras y mecanismos de acceso para dar tiempos de respuesta adecuados y compatibilidad en la información.

Software de traducción EDI

La tarea del software de traducción EDI, es la de convertir los formatos internos que utiliza la aplicación del organismo, a los estándares públicos para comunicación con otras compañías, así como el llevar a cabo el proceso inverso.

Existen tres tipos de traductores:

- los que funcionan sólo para recibir información, es decir sólo la imprimen;
- aquellos que envían y reciben pero como un "front end" para la aplicación;
- y los que pueden realizar comunicación de aplicación a aplicación.

Los requerimientos que debe satisfacer el software de traducción son:

- Manejo eficiente de los estándares EDI
- Conjuntos de transacciones completa
- Manejo de versiones actualizadas
- Interfases con las aplicaciones
- Habilidad de comunicaciones
- Facilidad de migración a otro sistema
- Manejado por pantallas o menús.

Software de Comunicaciones

Este software es el que permite enviar los datos utilizando la red de comunicaciones. Tiene el control de los protocolos a utilizar en la conexión con otras computadoras. Existen diferentes estándares para el manejo de las comunicaciones y estos deben estar contemplados en el software de comunicaciones a utilizar, se dividen de forma general en estándares Industriales, nacionales e internacionales. Algunos ejemplos son: la norma X.400 de CCITT (correo electrónico), la norma X.25 para redes conmutadas, el modelo de referencia de ISO, y los servicios mejorados a la X.400 llamados MHS/MOTIS (*message handling system/message oriented text interchange system*). Así mismo existen otras normas como ODA/ODIF (*office document architecture/office document interchange format*). Existen muchos productos que pueden trabajar en computadoras personales, y conectarse a los sistemas mayores. El protocolo más difundido en este tipo de equipos es el X/Y modem para comunicación entre micro-computadoras, aunque existen otros como el Kermit y el UUCP. Además se cuenta con protocolos propietarios como el XTALK o el PEP (*Packetized Ensemble Protocol*).

Sistema Traductor de Intercambio Electrónico de Datos

Unidades de Información

Las unidades de información base de los estándares se componen de los siguientes elementos:

- Elemento dato (la unidad de información identificable más pequeña posible; por ejemplo, un código de proveedor)
- Segmento de datos (varios elementos dato agrupados bajo un identificador. Ejemplo: un registro que contenga el número de la factura, el nombre de la misma, el rubro, la cantidad, etc.)
- Conjunto de transacciones (una secuencia de segmentos de datos que generan una transacción completa. Ejemplo: una factura con varios rubros)

- Grupo funcional (un grupo de conjuntos de transacciones del mismo tipo o de tipos similares entre sí. Ejemplo: un grupo de facturas con un grupo de órdenes de compra)

Elementos de Control

Los elementos de control se subdividen en:

Encabezados y caracteres de terminación
Delimitadores de campos
Conocimiento de recepción

En la figura 1.4 se muestra el flujo de una transacción en un sistema EDI. Como se puede observar, existe un requerimiento por parte de algún usuario que desea transmitir una transacción. Esto pasa a una aplicación de las que tiene la empresa que envía el mensaje, y de ahí pasa a una aplicación de interfase, la cual genera la transacción EDI, auxiliándose de la base de datos que contiene el

destino de la transacción, el tipo de transacción y toda la información de control necesaria. Al tener la transacción generada, esta se envía utilizando un módem, pasando por una red de comunicación, y llegando a otro módem, el cual deposita la transacción en la compañía destino, para que se lleve a cabo el proceso de traducción de la transacción, utilizando una base de datos EDI para validar que sea correcto el formato recibido. Después, la información es enviada

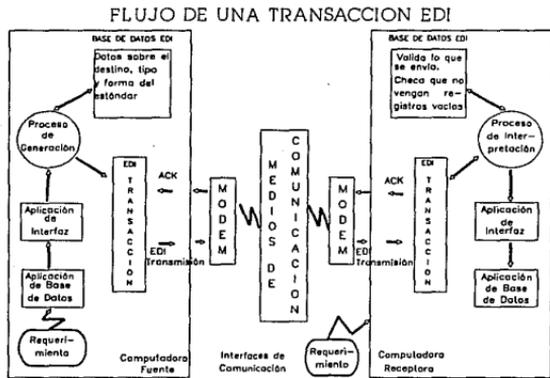


Figura 1.4. Diagrama de flujo de transacción EDI

por el programa de interfaz a la aplicación de la compañía receptora.

Sesión de Comunicaciones EDI

Las comunicaciones para manejar EDI han evolucionado de manera paulatina, empezando con conexiones directas, luego con una red de computadoras, varias redes interconectadas entre sí, y por último un sistema de correo prácticamente a nivel nacional o mundial. Existen comunicaciones punto a punto, multipunto y *dial up*.

Las características que se deben considerar en una comunicación punto a punto son: el *hardware*, las velocidades y el tipo de protocolos de comunicación, el tipo de producto que se utiliza, la ocupación de líneas, etc.

Por otro lado, al utilizar una red de valor agregado se debe estudiar lo siguiente: la seguridad, el compromiso de utilización, el costo, la capacidad para comunicarse con otras y el soporte que tenga la red.

1.1.4 Maneras de reaccionar en EDI, formas reactiva y promotora

Con EDI, las computadoras al poder comunicarse entre sí y hacer cálculos se convierten en herramientas valiosas para la competitividad, por lo que la mayoría de las compañías requieren este proceso para sobrevivir en la competencia de los mercados actuales. Un ejemplo de como afecta la supervivencia de las empresas es cuando un compañero comercial envía una CARTA obligando a su comprador o proveedor a utilizar los estándares EDI.

Esta CARTA viene típicamente de un gran usuario en la cual le informa a sus proveedores de productos o servicios, que después de un período de 3 a 6 meses, todas las transacciones hacia él deberán utilizar EDI o de lo contrario

cambiará de proveedor. Cuando entre estos usuarios se encuentran grandes compañías las pérdidas son cuantiosas.

Lo anterior ilustra las dos maneras de actuar al implantar un sistema EDI, la reacción y la promoción.

Cuando se promueve la implantación de un sistema EDI, se llevan a cabo las siguientes actividades:

- Plan global a nivel corporativo bien detallado
- Implantación controlada
- Adquisición de experiencia usando un programa piloto
- Plan de educación
- Comunicaciones con el compañero de pruebas
- Software de traducción
- Expansión

Al implantar EDI dentro del ambiente de una compañía se requiere hacer un examen de cual es su competencia, qué procedimientos complementa, qué utiliza y cómo posiciona a los usuarios con el resto de la compañía.

EDI reemplaza los sistemas de manejo de papel por un intercambio más básico, debido a que no puede reemplazarlos en una forma inmediata, deberá planearse su integración progresiva, lo que lleva a utilizar en ocasiones dos sistemas en forma simultánea.

EDI fuerza a que en una compañía con departamentos que utilizan diferentes sistemas de cómputo, el flujo principal de información sea distribuido al interconectarlos entre sí. Lo anterior obliga también a que existan cambios sustanciales en la forma tradicional de los flujos de información de negocios, patrones de trabajo y por lo mismo problemas.

Sistema Traductor de Intercambio Electrónico de Datos

EDI complementa y extiende la utilización de las bases de datos y la programación de aplicaciones. Así, EDI actúa como un comunicador entre las aplicaciones internas de la compañía y los archivos operativos de la misma (órdenes de trabajo, pagos, cuentas, compras, etc.) adicionalmente a la comunicación con entidades externas. Así que eventualmente EDI comunicará bases de datos de una manera suficientemente directa, con verificaciones de seguridad para el acceso y traductores en cada extremo (figura 1.5).

Por otro lado, cuando se tiene que reaccionar a la promoción, se presentan los siguientes problemas:

- Educación forzada
- Carencia de dirección en el proyecto
- Implantación apresurada
- Compañeros de intercambio forzados
- Necesidad de soluciones "rápidas"

COMPAÑIA A

COMPAÑIA B

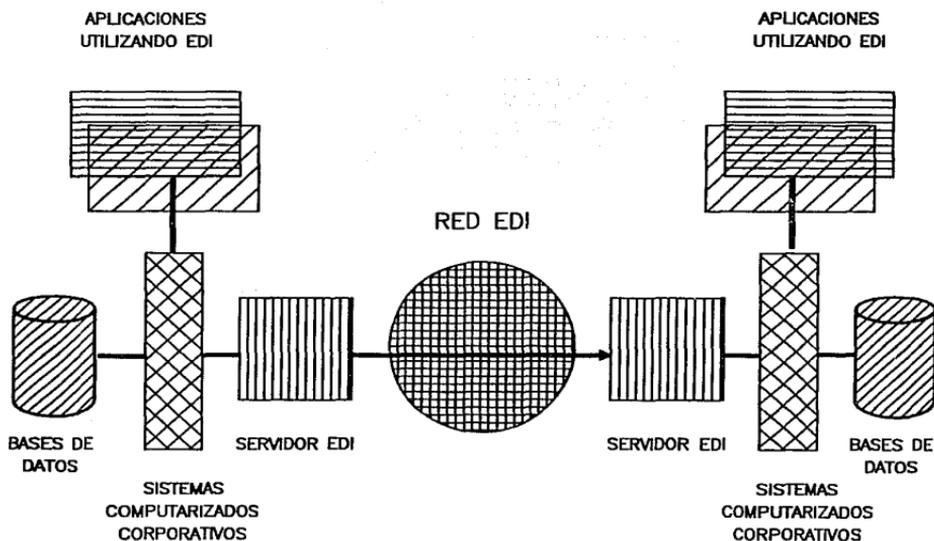


Figura 1.5. Comunicación externa entre dos compañías usando EDI

1.1.5 Madurez de la tecnología EDI

EDI es una tecnología joven. En la figura 1.6 se muestra una tabla comparativa de los formatos EDI con otras tecnologías en uso en la mayoría de las compañías de hoy. El factor limitante de mayor importancia son los estándares, los cuales controlan la conectividad.

Se entiende como tecnología base aquella que es imprescindible en un medio ambiente de negocios de nuestros días.

La tecnología clave es la que es requerida sólo en algunas ramas específicas, un ejemplo es el teleproceso, que no es un recurso crítico en todos los casos.

Las pruebas Figura 1.6 Diagrama comparativo de tecnologías piloto se llevan a cabo casi siempre en los departamentos de investigación de compañías o de universidades, con la finalidad de poner a prueba aquellas tecnologías que pueden ser de utilidad.

Por último, la emergente se da en el ámbito académico o en los organismos que rigen una determinada rama industrial.

En la telefonía se marcaron los estándares para manejo de voz desde sus inicios, y recientemente se utilizó la tecnología de tonos para el manejo de datos, un ejemplo claro de estándares para conectividad, es el uso del FAX.

La computación llegó más rápidamente a ser una tecnología base, debido a la enorme reducción de carga de trabajo para el ser humano, y a los beneficios económicos que se desprenden de su utilización.

Los sistemas de manejo de oficina comienzan a ser tecnología base en los países industrializados, debido a que

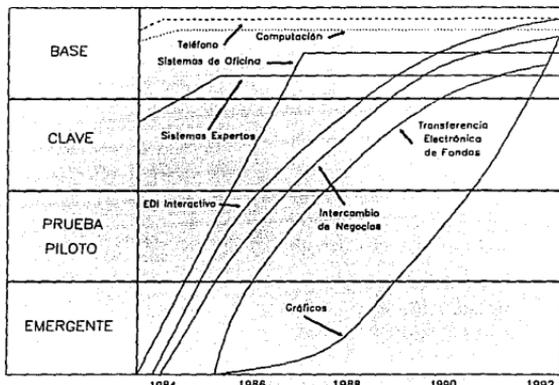
el flujo de información dentro de las empresas debe ser cada día más ágil por cuestiones de competitividad.

Los sistemas expertos están siendo introducidos rápidamente en las industrias de manufactura y en los ámbitos de investigación y de diseño de

productos. Un ejemplo es el Diseño Asistido por Computadora (CAD), el cual es imprescindible en muchas industrias hoy en día.

Existen tres grandes formas de EDI:

- Intercambio de datos comerciales (TDI Trade Data Interchanges)
- Transferencia electrónica de fondos EFT (Electronic Funds Transfer).
- EDI interactivo e intercambio gráfico de documentos.



Sistema Traductor de Intercambio Electrónico de Datos

Las diferentes fases de madurez tecnológica de las ramas de EDI se muestran en la figura 1.6. EFT es el líder entre las redes EDI y es la base de la tecnología para transferencia de fondos e Información entre Instituciones de crédito y bancarias.

Las formas de TDI (facturas, ordenes y notas de entrega) existentes para las Industrias que usan EDI están introduciéndose rápidamente como base para la industria manufacturera. Su implantación se está cristalizando en dos formas principales:

La primera es una implantación privada para grandes compañías con sus clientes y proveedores usando su red privada.

La segunda, que empieza a ser dominante es una red de Servicios de Valor Agregado (VANS *Value Added Network Services*), compartida entre varios usuarios de diferentes sectores.

En otros sectores el EDI Interactivo esta aún en programas piloto. Las gráficas y el diseño cooperativo esta emergiendo como base en las industrias de diseño (Automotriz, Aeronáutica), donde la concepción está siendo promocionada más y más entre los proveedores, y los sistemas de Diseño Asistido por Computadora (CAD) interactúan utilizando transferencia de archivos entre compañías .

En Europa, EDI para TDI y gráficas llegará a ser clave en uno o dos años, mientras que en Estados Unidos ya se está entrando en esta fase. Esto implica que la tecnología es utilizada por compañías líderes como una característica competitiva. En ciudades y sectores específicos de Inglaterra, como el sector distribución, están bastante avanzados.

Algunos sectores dominantes están formando sus propias redes EDI usando asociaciones industriales. Como ejemplos tenemos al sector químico con CEFIC (*Conseil Europeen des Federations de Chimie*), con su propia red, y a la Industria automotriz con ODETTE (*Organization for Data Exchange Through Teletransmission in Europe*). En estos casos, han sido los usuarios y no los proveedores, quienes han tomado la iniciativa y han puesto las bases y administración por medio de las organizaciones comerciales.

Las principales barreras para el crecimiento en el manejo de los estándares son:

- La difusión lenta de la aceptación y credibilidad de la tecnología de la información entre las empresas y el sector. El papel ha sido empleado por cientos de años y es difícil desplazarlo.
- Existen muy pocos estándares comunes para operación universal. Estos estándares son la base de la conectividad y del crecimiento inmediato, y determinan que compañía puede comunicarse con quién y a que costo.

Esta última razón es la más difícil de superar. En ausencia de estándares internacionales se han desarrollado más los estándares nacionales; en Estados Unidos en el ANSI, los estándares X.12 son las reglas a seguir; mientras que en Europa cada país tiene las suyas.

Sin embargo, avances recientes, promovidos por las Naciones Unidas en la reunión de la UNECE y JEDI (ANSI X.12 EDI committee), dieron como resultado el respaldo de EEC (*European Economic Commission*) a los estándares EDIFACT (*EDI for Administration, Commerce and Transport*), para los negocios internacionales. Ahora se presenta un conflicto creciente entre los estándares de los

Estados Unidos y los Internacionales, ya que cada líder industrial afirma que sus estándares son los mejores para establecerlos como internacionales.

1.1.6 El lugar de EDI en la estructura de negocios

EDI coloca a las compañías en una nueva posición de negocios tanto con clientes como con proveedores. El departamento de procesamiento de datos (DPD) normalmente maneja transacciones EDI y llega a ser la ventana de negocios al mundo de la compañía.

En ocasiones, el departamento usuario trata directamente con EDI, pero muchas compañías prefieren hacerlo a través del DPD por ser los que más experiencia tienen en cómputo y comunicaciones. Aún las compañías externas de comunicaciones deben pasar a través del DPD.

Por eso, el papel del DPD se vuelve crucial, por ser el enlace técnico entre empresas. Más aún, este departamento se vuelve la cara comercial presentada a clientes y proveedores, lo que aumenta la carga de trabajo para este departamento, ya que ahora es responsable del manejo de todo el "papeleo" de negocios. El departamento y sus redes se ponen a trabajar en un modo basado en transacciones en semi-tiempo real que no se ha visto hasta hoy en los negocios. Se requiere expansión y mejoras para asegurar que esta operación sea exitosa.

Una vez que el sistema se ha instalado, los errores de captura se reducen, así como las fallas en documentos, especialmente con el uso de reconocimiento de recepción automática.

EDI está emergiendo como una herramienta comercial de varias formas, siendo la principal de ellas la imagen de la compañía. Los usuarios han encontrado que esta imagen se incrementa con EDI en cuanto a eficiencia y confiabilidad.

1.1.7 EDI y sus interfaces con aplicaciones existentes

Existen dos niveles de interfaz para EDI (figura 1.8). Primero, la interfaz de la compañía saliendo a través de la red hacia sus contrapartes externas. En segundo lugar está la interfaz interna, del centro de cómputo con el resto de la compañía. Aún cuando el primero parece ser el más difícil, el segundo es el más importante, tanto para lograr su utilización en forma general, así como el apoyo del resto del personal y sobretodo debe resolverse la interconexión entre mainframes, minicomputadoras y computadoras personales de diferentes tipos, de tal forma que se cambia el concepto de interconectar procesos computarizados por

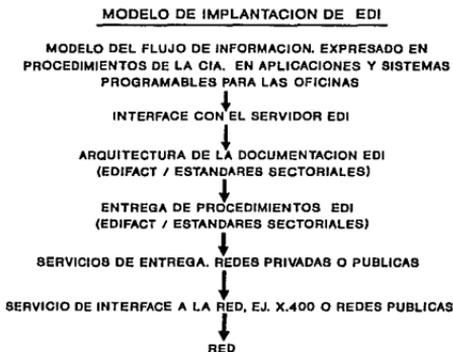


Figura 1.8. Interfaces de un sistema EDI

medio de Interfaces de papel por interfaces electrónicas.

Muchas compañías han resuelto la interfaz externa como parte de proyectos piloto utilizando transferencia de archivos con discos flexibles o recapturando manualmente la información en una computadora personal.

Sistema Traductor de Intercambio Electrónico de Datos

1.1.8 Direcciones futuras

La tecnología EDI está en desarrollo todavía, lo que significa que explotará ampliamente los desarrollos en arquitecturas de redes de comunicación, manejadores de bases de datos, sistemas de automatización de oficinas y quizá, más que todo lo anterior, los de Inteligencia Artificial en su avance en los sistemas expertos para tareas administrativas.

El problema técnico más grande que tiene que resolver EDI es el formatear mensajes de manera estándar. La mayoría de las empresas ya tienen contabilidad y sistemas de inventario en base a servicios computarizados, así que mandar una factura o orden de compra de una compañía a otra en forma electrónica parece un proceso natural. La realidad es que la mayoría de las bases de datos no están diseñadas para que puedan dialogar entre sí, y aún cuando algunas sí pudiesen hacerlo, existe renuencia a permitir que entidades externas accedan información que es considerada crítica.

EDI utiliza formas estándares y filtros especializados. Cuando es necesario enviar una orden de compra o una factura, un programa filtro extrae la información apropiada de la base de datos de la compañía remitente y la traslada a un formato EDI, éste es transmitido utilizando cualquier medio de comunicaciones a la compañía receptora. En ésta la forma recibida debe ser trasladada y filtrada en un proceso inverso para que la información sea depositada en su base de datos.

Al utilizar este tipo de procedimientos las compañías garantizan una transferencia de información limpia de errores y disponible rápidamente, en lugar de un proceso lento y vulnerable a los errores de recaptura.

Al ser EDI una versión especializada de correo electrónico no es extraño que sus estándares hayan chocado con el estándar X.400 "e-mail" de la ISO, por lo que usuarios de ambos están forzando la utilización de la norma X.400 como envoltura de EDI. Se espera que a finales de 1990 la CCITT logre sacar la recomendación P.EDI, pero aún así podría tomar dos años antes de que los usuarios puedan aprovechar esta alternativa de solución.

1.1.9 Puntos legales en EDI

Para establecer un compañero de EDI, es necesario revisar los acuerdos legales, así como los controles de auditoría y seguridad, los términos y condiciones bajo los cuales se realizará el intercambio de información.

Los acuerdos entre compañeros de negocios deben contemplar: los límites legales del acuerdo; los estándares de los formatos a intercambiar; los procedimientos para establecer la comunicación; el formato que se transmitirá; la red a utilizar; y por último los errores y la protección contra fraude.

Para una mayor información sobre tópicos legales en EDI se puede consultar el libro "Legal Issues Impacting EDI", editado por la TDCC/EDIA.

1.2. PROBLEMATICA

Como se puede observar prácticamente todas las industrias en los Estados Unidos y muchas en Europa tienen actualmente al menos un miembro utilizando EDI. Por lo anterior, es importante que las empresas ubicadas en la república Mexicana se preparen para trabajar con los formatos EDI, debido a que sostenemos relaciones comerciales muy estrechas con compañías extranjeras, la prueba de ello se vive actualmente en Ferrocarriles Nacionales de México.

Introducción

Las relaciones de los Ferrocarriles Nacionales de México (FNM) con la industria ferroviaria de los Estados Unidos y Canadá son reguladas por la Asociación Americana de Ferrocarriles (AAR Association of American Railroads), la cual está formada por varios comités, en los cuales se deciden asuntos de orden general que afectan a esta industria.

Un ejemplo de las relaciones entre FNM y los ferrocarriles extranjeros es el pago por renta de carros. Cuando un ferrocarril recibe un carro de tren que no es de su propiedad, éste debe pagar una renta, llamada renta de carros (Car Hire), al propietario de la unidad, considerando el tiempo de estancia en las líneas del ferrocarril receptor (perdiem) y las millas recorridas sobre sus vías (millaje).

El comité de Renta de carros (COCH Comité on Car Hire) es el que se encarga de controlar el proceso de pagos por concepto de perdiem y millaje entre los diferentes ferrocarriles.

En la actualidad FNM utiliza un proceso semi-automatizado para generar el pago de la renta de carros a los ferrocarriles extranjeros, utilizando formatos en papel, manejando un volumen aproximado de 20,000 carros al mes.

La AAR tiene un sistema que ya maneja el envío y recepción de información utilizando cintas magnéticas para el pago de la renta de carros, pero este sistema utiliza formatos de longitud fija; y el COCH desarrolló requerimientos de datos durante los últimos dos años para revisar las carencias del sistema actual de la AAR.

Después de realizar un estudio sobre los datos que se intercambian actualmente en el sistema de cintas y formato fijo, se notaron algunas carencias, por lo que el COCH hizo el planteamiento sobre los datos que eran necesarios para subsanar estas carencias.

Fruto de estos requerimientos, se generó una regla que obliga a todos los ferrocarriles miembros de la AAR (FNM es miembro), a intercambiar la información de renta de carros de manera electrónica, utilizando formatos EDI, e imponiendo una multa de un dólar (1 ds.) por cada carro que se envíe usando formatos en papel.

A partir de 1988 la AAR determinó que todos los nuevos sistemas o formatos computarizados se desarrollaran utilizando EDI, debido a su gran flexibilidad para adicionar o modificar campos, además del ahorro al intercambiar únicamente la información necesaria.

El comité de Sistemas de Datos de Contabilidad de la AAR desarrolló el formato de cómputo requerido utilizando estándares EDI, los cuales entrarán en vigor el primero de junio de 1991.

Por lo anterior, el FNM debe desarrollar un sistema que maneje estándares EDI para obtener los beneficios que implican el usar Intercambio electrónico de Información y evitar el pago de la multa de un dólar por carro extranjero que se interna en México.

1.3 OBJETIVO Y ALCANCES

El objetivo del trabajo de tesis es desarrollar un sistema de traducción de formatos EDI, que funcione como un preproceso ("front end") para aplicaciones, que contemple 8 formatos públicos que serán utilizados en Ferrocarriles Nacionales de México.

esta página fué dejada en blanco intencionalmente

CAPITULO 2

análisis del sistema

2.1 ANALISIS DEL SISTEMA

El Análisis del Sistema es un proceso sistemático de razonamiento sobre un problema y las partes que lo constituyen. Su objetivo es obtener una *especificación de requerimientos*, es decir, que se necesita o que debe ser hecho.

Este proceso envuelve la comunicación con mucha gente:

Personas familiares con la necesidad, personas que trabajan para satisfacerla, personas que pagan por la solución y personas que trabajan con el problema.

Los productos principales generados son el *Diseño Conceptual del Sistema de Información* y la *Propuesta de Implantación*, estos productos son vitales para el éxito del proyecto.

El proceso de análisis debe cubrir las siguientes características:

- 1) Metodología de análisis apropiada
- 2) Localización fácil y rápida de la información
- 3) Soporte de presentación y discusión para facilitar la comunicación

A continuación se desglosan los pasos seguidos durante el proceso del análisis realizado para el presente trabajo.

2.2 ALCANCES

Los Alcances del Sistema describen los servicios básicos que debe proveer la aplicación, así como un marco de referencia acerca de la amplitud del proyecto.

El sistema deberá captar y procesar datos para intercambio electrónico entre diversas compañías y constará de los siguientes procesos:

- Recepción de información de la compañía.
- Recepción de información EDI
- Traducción de formatos de la compañía a formatos EDI
- Traducción de formatos EDI a formatos de la compañía
- Formatos EDI restringidos de acuerdo a convenios entre los compañeros

2.3 CARACTERISTICAS Y BENEFICIOS

Las Características son frases que describen los puntos particulares de la aplicación y que identifican aspectos especiales.

Puede ser conveniente separarlas en características técnicas y funcionales.

El sistema contará con las siguientes características:

- Ser de fácil manejo, para que pueda ser utilizado por personal sin experiencia en computadoras
- Mantener en forma consistente la información durante todo el proceso
- Ser flexible a modificaciones en la información a intercambiar
- Estar protegido contra accesos no autorizados

Sistema Traductor de Intercambio Electrónico de Datos

- Ser Independiente del *Hardware* en que se ejecute

A continuación se describen los Beneficios que el sistema aportará, esta sección es importante en la venta del producto. Los beneficios pueden ser cuantificables (costos de operación) y no cuantificables (satisfacción personal).

El sistema aportará los siguientes beneficios:

- Reducir el manejo de papel entre compañías
- Evitar la recaptura de la información
- Disminuir los errores de captura
- Reducir los ciclos de ordenes
- Procesar un mayor número de documentos en menor tiempo
- Agilizar el trámite de algunos pagos entre FNM y sus clientes
- Evitar el pago de multa a los FFCC americanos por manejo de papel

2.4 USUARIOS

El Usuario es el receptor final del sistema, es el responsable de decidir la integración del mismo a sus actividades. Existen dos tipos de usuarios:

Directos :

Operador : Administrador o empleado de un área que utilizará el producto.

Indirectos :

Propietario : Solicitante del proyecto, asigna personal de su área al proyecto y se queda con la administración del mismo.

Cliente : Director del área que aprueba y reprueba el proyecto.

El usuario es el experto en el área y la solución debe salir de él, apoyado en la información proporcionada por el analista.

A continuación se muestran los usuarios identificados en el sistema :

Directos

Operador : Manejo y control del sistema

Indirectos

Gerencia de Logística de carros de FNM: Generación y manejo de información (renta de carros).

Gerencia de Sistemas de FNM: Generación y manejo de información automatizada (renta de carros).

Subdirección General de Finanzas de FNM: Generación y manejo de información automatizada (facturación).

2.5 DEFINICIONES Y TERMINOS ESPECIALES

Para una mejor comprensión de los términos manejados en el presente trabajo se realiza a continuación una definición de éstos.

Estándar EDI

Conjunto de reglas, agrupadas, aceptadas y adheridas voluntariamente, por medio de las cuales los datos son estructurados en formatos de mensaje para el intercambio de información operacional y de negocios.

Elemento dato

Es la unidad de información más pequeña. Puede ser un solo carácter o una serie de caracteres constituyendo una descripción literal o una cantidad numérica. La longitud de un elemento dato puede ser fija o variable, pero consistente con los datos transmitidos.

Segmento de datos.

Un segmento de datos esta compuesto por un identificador de función y uno o más elementos de datos relacionados funcionalmente. Cada elemento dato está precedido por un delimitador y un carácter terminador inmediatamente después del último elemento dato transmitido.

Transacción.

Una transacción es un grupo de segmentos de datos estándar en una secuencia predefinida, necesarios para proporcionar todos los datos requeridos para definir una transacción completa.

Grupo funcional

Un grupo funcional está compuesto por un conjunto de transacciones del mismo tipo, o similares, transmitidas de la misma localidad y enmarcadas por un encabezado de grupo funcional y un segmento fin de fin de grupo funcional.

Formato EDI

Definición estándar de una transacción.

Protocolo de comunicación

Es la envoltura de los datos a ser transmitidos, utiliza un medio de comunicación como modem o controladores de comunicación.

LOOP

Iteración de un proceso dentro de un programa o de un dato dentro de una estructura de datos.

Compañero EDI

Compañía con la cuál se acuerda mutuamente efectuar transacciones utilizando estándares EDI.

2.6 ELEMENTOS DE UN SISTEMA EDI

Los elementos de un sistema EDI se dividen en dos:

FISICOS

Hardware

Computadora que genera el mensaje
Modem
Red de comunicaciones (pública o privada)
Computadora que recibe el mensaje

Software

Base de Datos fuente
Software de traducción
Software de comunicación
Base de Datos destino

LOGICOS

Unidades de información
Elementos de control
Sesión de comunicaciones

A su vez, las unidades de información base de los estándares se componen de los siguientes elementos (figura 2.1):

Elemento dato (la unidad de información identificable más pequeña posible, por ejemplo, un código de proveedor)

Segmento de datos (varios elementos de datos agrupados bajo un identificador, por ejemplo, un registro que contenga el número de la factura, el nombre de la misma, el rubro, la cantidad, etc.)

Conjunto de transacciones (una secuencia de segmentos de datos que generan una transacción completa, por ejemplo, una factura con varios rubros)

Grupo funcional (un grupo de conjuntos de transacciones del mismo tipo o de tipos similares entre sí, por ejemplo: un grupo de facturas con un grupo de órdenes de compra)

Los elementos de control se subdividen en:

Encabezados y caracteres de terminación
Delimitadores de campos
Conocimiento de recepción
Protocolo de comunicación

2.7 REQUERIMIENTOS

Los requerimientos son definiciones precisas de necesidades, describen características externas, o del medio ambiente visible al usuario, así como restricciones del medio. Describen la percepción de una necesidad y permiten la comunicación para comprender la solución.

La determinación de los requerimientos se logra a través de:

- 1) Analizar el problema
- 2) Documentar los requerimientos obtenidos

1 En los sistemas comerciales, este proceso se llama mapeo y es responsabilidad del usuario el adecuar los datos de la compañía a las reglas impuestas por el traductor. En nuestro caso, el análisis se desarrollará sobre un módulo que realice estas funciones, pero la implantación se apegará a un modelo comercial.

- 3) Verificar la precisión de la comprensión obtenida

Existen tres tipos de requerimientos:

- 1) Funcionales (análisis funcional)
- 2) Estructura de información (análisis de información)
- 3) No funcionales (restricciones existentes)

A continuación se muestran los requerimientos que el sistema debe cubrir.

REQUERIMIENTOS FUNCIONALES

El sistema deberá contemplar las siguientes funciones:
Recepción de la información de la compañía.

Aquí debe recibir la información proveniente de la base de datos de la compañía.

Recepción de la información EDI.

Aquí debe recibirse la información proveniente de las bases de datos del compañero EDI.

Traducción de formatos de la compañía.

Aquí deben traducirse los datos de las aplicaciones de la compañía a formatos EDI.1

Traducción de formatos EDI.

Aquí deben traducirse la información de los formatos EDI a formatos de la compañía.

Sistema Traductor de Intercambio Electrónico de Datos

ESTRUCTURA DE INFORMACION

Hasta el momento no es posible definir la estructura de la información, debido a que no existe un sistema previo de intercambio electrónico de información.

REQUERIMIENTOS NO FUNCIONALES:

- Turbo Pascal 5.0
- Dbase III Plus
- PC 640K /1 puerto serie
- Modem
- Líneas telefónicas
- Deberá estar terminado dos meses después de concluir el análisis.

2.8 DIAGRAMA CONCEPTUAL DE USUARIO

Uno de los principales problemas en el desarrollo de sistemas es establecer una comunicación sencilla y efectiva entre el analista y el usuario, para lograr la comprensión total del sistema.

Los diagramas conceptuales de usuario son una herramienta útil y sencilla que facilita la exposición gráfica de lo que será el sistema, según la concepción del analista, permitiendo la interacción entre analista y usuario.

Para la generación de estos diagramas se debe dividir el sistema en puntos funcionales, por motivos de claridad se utilizan varios niveles de diagramación, que van de un contexto general a niveles de detalle.

En el diagrama de contexto se presenta a todo el sistema como una sola función, especificando sus entradas y salidas.

En los siguientes niveles se va incrementando el grado de detalle, de manera que se va desglosando cada proceso general en subprocesos. En el caso de que uno de estos subprocesos resulte muy complejo, se baja a un siguiente nivel y así sucesivamente.

Durante la generación de los diagramas existen varios puntos a considerar, que son los siguientes:

- Demasiados niveles de diagramas son difíciles de leer.
- Cada diagrama conceptual debe tener entre 3 y 7 procesos.
- Un proceso con actividades distintas es candidato a descomponer.
- Los procesos de bajo nivel deben ser descritos en forma concreta y breve.

La simbología utilizada en la diagramación se muestra en la figura no. 2.2

2.8.1 Sistema Traductor de Formatos EDI

Diagrama Conceptual de Usuario

de negocios, la creación y adecuación de la transacción y el control de transmisión. Como involucra varias actividades fue expandido a primer nivel (fig. 2.4).

En la figura no. 2.3 se muestra el diagrama de contexto general del sistema.

El sistema se encuentra dividido fundamentalmente en 5 funciones, las cuales se detallan a continuación.

1. Recepción de Información de la Compañía.

La función desempeñada es la de obtener datos pertenecientes a la compañía y que se desea sean traducidos a formatos EDI, para su posterior transmisión a un compañero de negocios. Dicha obtención de datos se puede realizar ya sea por captura o por extracción de la base de datos de la compañía. Como sólo se realiza la actividad de obtención no se consideró para ser expandido a un primer nivel de detalle.

2. Recepción de Información EDI.

Obtiene transacciones EDI provenientes de los compañeros de negocios de la base de datos EDI. Esta función tampoco fué considerada para ser expandida.

3. Traductor de Formatos de la Compañía.

Su función es la de crear transacciones EDI, para lo cual realiza diversas actividades como son la identificación de la compañía y del compañero

SIMBOLOGIA DEL DIAGRAMA CONCEPTUAL DE USUARIO

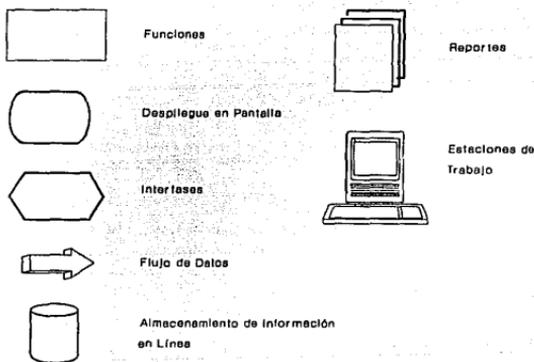


Figura 2.2 Simbología del diagrama conceptual de usuario

Sistema Traductor de Intercambio Electrónico de Datos

Las actividades en que fue desglosado son: La identificación de los formatos de la compañía, la cual obtiene esta información de la tabla de formatos; la identificación de los compañeros EDI, para conocer los datos generales del compañero, así como el caracter delimitador que utiliza. Una vez identificados estos datos se procede a la Creación de la transacción EDI, quien accesa la base de datos de la compañía para obtener los datos a traducir. Posteriormente, se lleva a cabo la Adecuación de la transacción EDI, para escribirla en la correspondiente base de datos; de donde la Transmisión de la información obtiene la transacción para controlar su transmisión.

cual se muestra en la figura no. 2.5

SISTEMA TRADUCTOR DE FORMATOS EDI

(Diagrama conceptual de usuario)

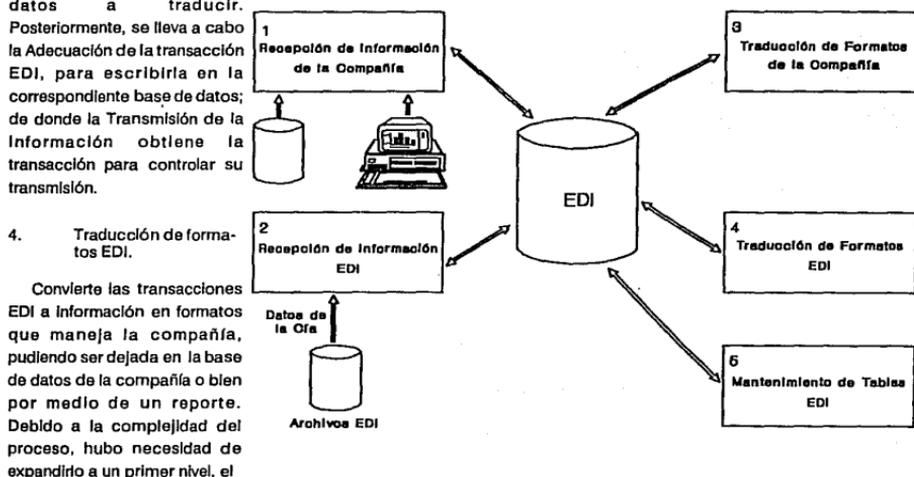


figura 2.3. DFD de segundo nivel del proceso de Traducción a Información de la Compañía

Las actividades que realiza a un primer nivel de detalle son: El Análisis de Grupo Funcional, quien debe consultar la base de datos EDI, para obtener tanto los datos del compañero como el tipo de las transacciones que envuelve el grupo funcional. Con estos datos se procede a la Traducción de la Transacción EDI, quien requiere de acceder las tablas de traducción para poder generar información en formatos de la compañía. Posteriormente se genera una Transmisión de Acuse de Recibo al compañero de negocios, para que esté enterado de que ya se recibió la transacción.

5. Mantenimiento de Tablas EDI.

Actualiza las tablas EDI, realizando altas, bajas y cambios de transacciones, segmentos, elementos dato, compañeros de negocios y formatos de la compañía. No se consideró para ser desglosado a un siguiente nivel.

TRADUCCION DE FORMATOS DE LA COMPAÑIA

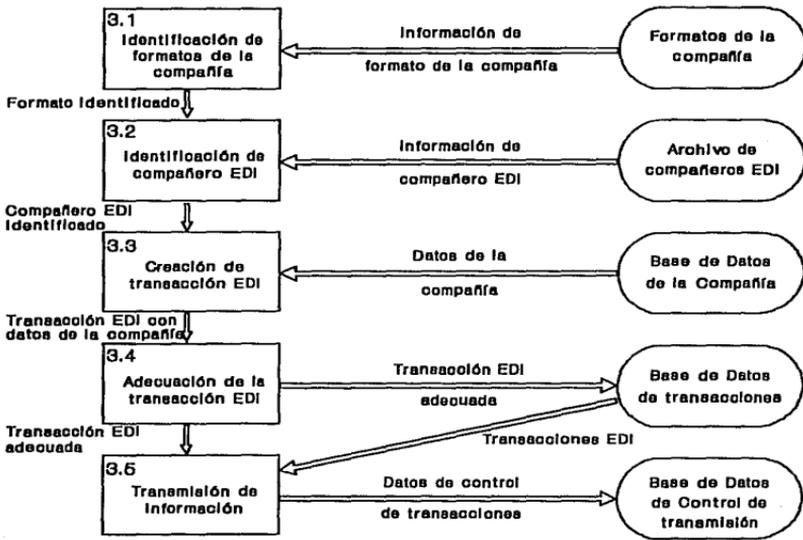


Figura 2.4 Diagrama de Flujo de Primer Nivel

TRADUCCION DE FORMATOS EDI

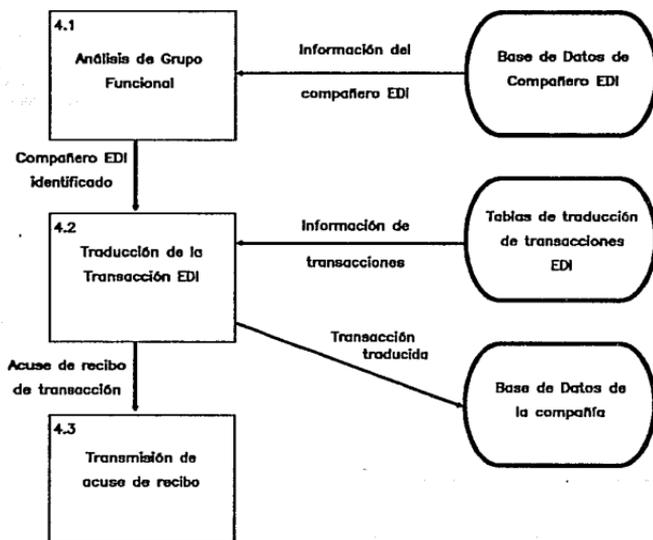


Figura 2.5 Diagrama de Primer Nivel de la función 4

Sistema Traductor de Intercambio Electrónico de Datos

2.9 DIAGRAMA DE FLUJO DE DATOS.

Un Diagrama de Flujo de Datos (DFD), es una técnica gráfica que describe el flujo de información y las transformaciones que se aplican a los datos, conforme se mueven de la entrada a la salida.

El DFD puede usarse para representar un sistema a cualquier nivel de abstracción. De hecho, los DFD pueden partitionarse en niveles que representan flujo incremental de información y detalle funcional. Un nivel de contexto general, también llamado un modelo de sistema fundamental, representa el elemento de software entero como una única burbuja con datos de entrada y salida, indicados por flechas hacia dentro y hacia afuera, respectivamente. Transformaciones adicionales y caminos de flujo de información se representan, conforme el nivel de contexto general es subdividido para revelar más detalles.

La simbología de un DFD es la siguiente:

- Se utiliza un rectángulo para representar una entidad externa, esto es, un elemento del sistema u otro sistema que produce información que ha de ser transformada por el software o que recibe información producida por el *software*.
- Un círculo o burbuja representa un proceso o transformación que se aplica a los datos y que los cambia de alguna forma.
- Una flecha representa uno o más elementos de datos. Todas las flechas de un DFD deben estar etiquetadas.

- Un pequeño rectángulo sin cerrar representa un almacenamiento de datos - información almacenada que es usada por el programa.

El diagrama de flujo de datos es una herramienta gráfica que puede ser muy útil durante el análisis de requerimientos. Sin embargo, el diagrama puede causar confusión si su función se confunde con la de un diagrama de flujos. Un diagrama describe el flujo de la información sin una notación explícita del control, por ejemplo, condiciones o iteraciones.

Unos cuantos criterios sencillos pueden ayudar inmensurablemente durante la desviación de un flujo de datos orientada al software:

- 1) El nivel de contexto general debe describir al *software* como una burbuja sencilla.
- 2) Los archivos de entrada/salida principales deben ser anotados cuidadosamente.
- 3) Todas las flechas y círculos deben estar etiquetadas (con nombres con significado).
- 4) La continuidad del flujo de información debe ser mantenida.
- 5) Cada vez debe refinarse una burbuja, expandirla en sus diferentes funciones.

2.9.1 Descripción de Diagramas de Flujo de Datos.**NIVEL 1**

En la figura 2.6 se muestra el diagrama de flujo de datos de nivel 1, el cual está formado por los siguientes procesos:

1. Recepción de Información.
Obtiene información de los formatos de la compañía, así como de las transacciones EDI que provienen de las compañías, con las que se tiene intercambio electrónico de datos (compañeros EDI).
2. Traducción a Información EDI.
Traduce los formatos de la compañía a transacciones EDI, auxiliándose de la base de datos EDI.
3. Transmisión
Controla el estado de las transmisiones realizadas, de manera que, cuando existe un error, emite petición de retransmisión. Registra la fecha y los datos del emisor y del receptor de la transacción.
4. Traducción a Información de la Compañía.
Traduce las transacciones EDI a formatos que maneja la compañía.
5. Depósito de la Información.
Registra en la base de datos de la compañía la información ya traducida de las transacciones EDI.
6. Genera Reporte.

Produce el reporte de cada transacción traducida a información de la compañía.

7. Mantenimiento de Tablas.

Actualiza las tablas de la base de datos EDI, que son utilizadas para la realización de cualquier traducción.

NIVEL 2

En la figura 2.7 se muestran el segundo nivel del diagrama de flujo de datos del proceso de Recepción de Información.

A continuación se presenta una breve descripción de cada una de las funciones que realizan los procesos:

1. RECEPCION DE INFORMACION.

- 1.1 Recepción de Información EDI.
Obtiene transacciones de la base de datos de mensajes EDI que fueron enviadas por compañeros EDI.
- 1.2 Recepción de Información de la Compañía.
Extrae información de la base de datos de la compañía de acuerdo al tipo de transacción que se desee generar y la registra en el archivo *buffer* de la compañía.
- 1.3 Captura de Información de la Compañía.
Captura información de acuerdo al tipo de transacción a generar.

Sistema Traductor de Intercambio Electrónico de Datos

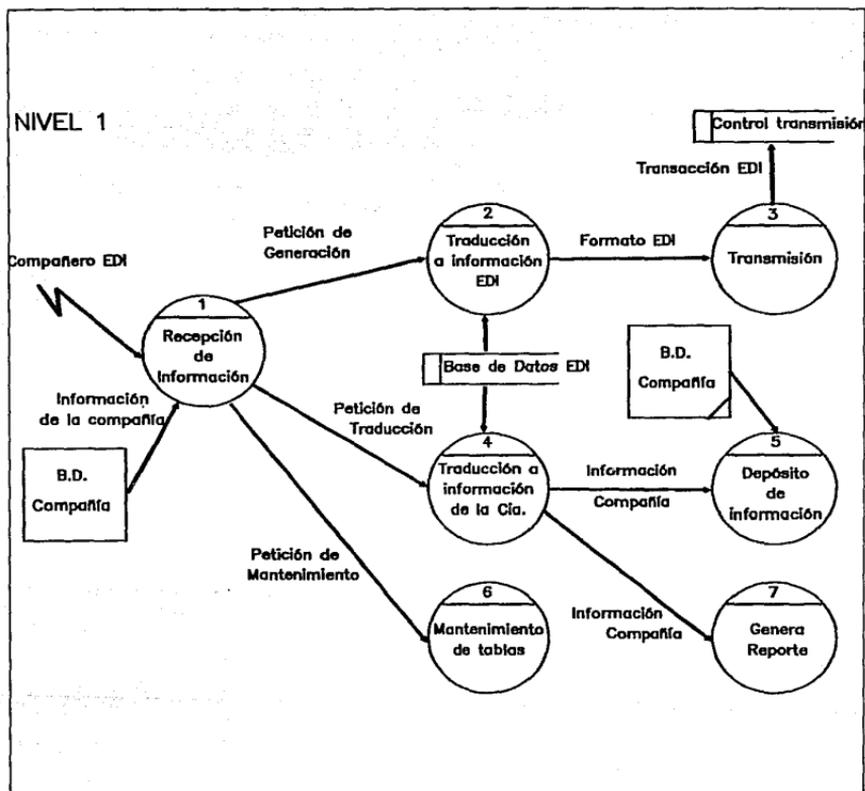


Figura 2.6 Diagrama de Flujo de Datos de primer nivel

análisis del sistema

2. TRADUCCION A INFORMACION EDI

En la figura 2.8 se muestran el segundo nivel del diagrama de flujo de datos, del proceso de Traducción a Información EDI.

A continuación se presenta una breve descripción de cada una de las funciones que realizan los procesos:

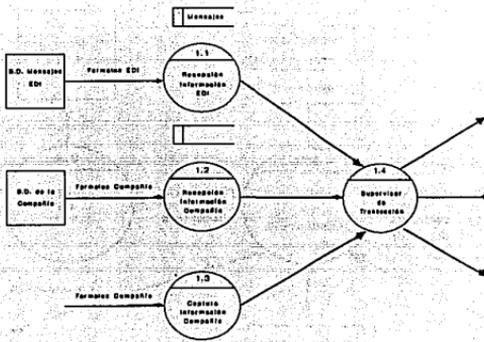


Figura 2.7 DFD de segundo nivel. (Recepción de información)

1.4 Supervisor de Transacción.

Controla el flujo del programa de acuerdo al tipo de movimiento que se desea realizar, efectúa la generación o traducción de transacciones o mantenimiento de tablas.

2.4

2.1 Identifica Formato de la Compañía.

Obtiene del archivo de formatos de la compañía que tipo de formato es el que se utiliza para la transacción a generar, registrándolo en un archivo de área de trabajo.

2.2 Identifica Compañero EDI.

Obtiene del archivo de compañeros EDI los datos necesarios del compañero al que se va a enviar la transacción. Proporcionando los datos particulares como son nombre, identificador, clave de acceso, carácter que utiliza como delimitador, etc. Graba esta información en una área de trabajo.

2.3 Creación de Transacción.

Forma la transacción auxiliándose de la base de datos EDI y los archivos *buffer* de la compañía y área de trabajo.

La base de datos EDI está formada por los siguientes archivos: Nombres de Conjuntos, Segmentos, Nombre de Segmentos, Elementos del Conjunto y Elementos Datos.

Escribe en EDI.

Registra en la base de datos de mensajes EDI la transacción generada.

Sistema Traductor de Intercambio Electrónico de Datos

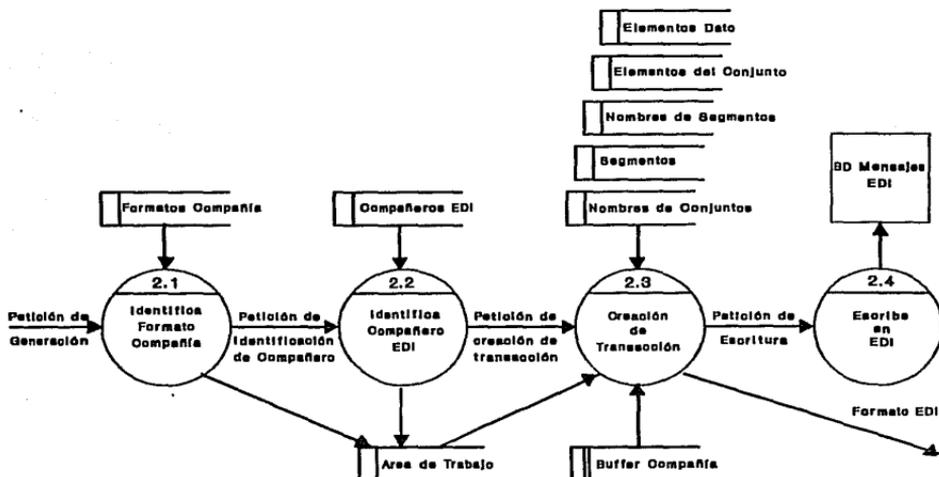


Figura 2.8 Diagrama de flujo de segundo nivel del proceso Traducción a Información EDI

4. TRADUCCION A INFORMACION DE LA COMPAÑIA

En la figura 2.9 se muestran el segundo nivel del diagrama de flujo de datos del proceso de Traducción de Información de la Compañía.

A continuación se presenta una breve descripción de cada una de las funciones que realizan los procesos:

4.1 Validación de la Clave de Acceso.

4.2

Aceptación o Rechazo.

Genera transacción de aceptación o rechazo, según el requerimiento que reciba, escribiéndola en el archivo *buffer* de transacciones.

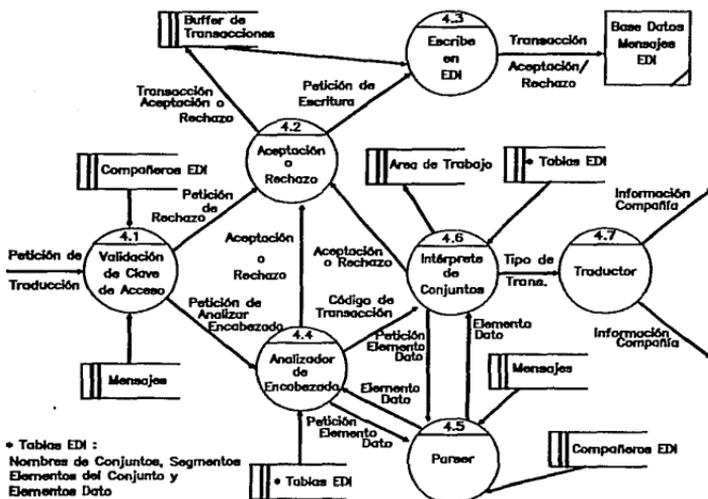


figura 2.9 Diagrama de flujo de datos de segundo nivel del proceso de Traducción a Información de Compañía

- | | | |
|---|--|---|
| <p>4.3 Escribe en EDI.</p> <p>Obtiene el tipo de la transacción del archivo <i>buffer</i> de transacciones y lo graba en la base de datos de mensajes EDI.</p> | <p>4.4 Parser.</p> <p>Identifica cual es el carácter que se utiliza como separador en la transacción , auxiliándose con el archivo de compañeros EDI.</p> | <p>4.5 Analizador de Encabezado.</p> <p>Analiza los dos diferentes encabezados de control, por lo que requiere llamar iterativamente al módulo parser para que le proporcione elementos dato, los cuales debe verificar contra la base de datos EDI. En caso</p> |
| <p>4.3 Escribe en EDI.</p> <p>Una vez identificado el caracter separador proporciona un elemento dato cada vez que es requerido.</p> | | |

Sistema Traductor de Intercambio Electrónico de Datos

de error genera petición de rechazo de transacción, de lo contrario cede el flujo del programa al módulo intérprete de conjuntos.

4.6 Intérprete de Conjuntos.

Verifica que cada elemento dato tenga las condiciones y el orden establecidos en la base de datos EDI, genera la petición de rechazo de transacción en caso de error.

4.7 Traductor.

Convierte una transacción a formato interno de la base de datos de la compañía.

2.10 DESCRIPCIÓN DE PROCESOS

Una vez elaborados los Diagramas Conceptuales de Usuario, y ayudados por los Diagramas de Estructura de Información, se procede a definir los procesos con base a los requerimientos funcionales (operaciones básicas a realizar por función).

Los procesos deben tener las siguientes características:

Verificables

- Debe ser posible verificar si el sistema implantado cumple con los requerimientos del usuario.
- Debe contener un parámetro para evaluar el éxito del desarrollo.
- Los requerimientos más concretos son más fáciles de verificar, pero el ser específico puede coartar la libertad del diseñador.

Consistentes

- Si los requerimientos son inconsistentes es obvio que NO PODRAN SER IMPLANTADOS.
- Cuando se revisa este punto es necesario efectuar los siguientes razonamientos:
¿Es posible obtener las salidas con base a las entradas y procesos definidos?
¿Los diagramas de alto nivel concuerdan con los diagramas de bajo nivel?
¿Los agregados de información, registros y elementos de DCU son consistentes con los requerimientos?

No redundantes

- La redundancia de requerimientos causa problemas de consistencia en el desarrollo y mantenimiento.

Puntos a incluir en las descripciones de procesos

Nombre.- Debe ser único.

Frecuencia.- Con qué frecuencia se ejecuta el proceso.

Objetivo.- Indica al lector el propósito del mismo.

Operaciones básicas.- Descripción de los estados internos del proceso.

Entradas, Salidas, Entidades.- Relación del proceso con su medio ambiente.

DESCRIPCION DE PROCESO

| | | |
|---|----------------|---------------------------|
| SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS. | | FECHA 04/DIC/90 |
| PROCESO 1.0 Recepción de Información de la compañía | | |
| OBJETIVO Obtener la información en el formato que maneja la compañía para traducirla a estándares EDI. | | |
| FRECUENCIA Diario. Tantas veces como se requiera enviar información a otras compañías. | | |
| OPERACIONES BASICAS <ul style="list-style-type: none">● Capturar la información a transmitir u obtenerla de la base de datos de la compañía. | | |
| ENTIDADES ACCESADAS TEBDCA. Base de Datos de la Compañía. | | |
| ENTRADAS Formatos de la Compañía. (ver descripción de entradas-salidas) | SALIDAS | |

DESCRIPCION DE PROCESO

| | | |
|--|----------------|---------------------------|
| SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS. | | FECHA 04/DIC/90 |
| PROCESO 2.0 Recepción de información EDI | | |
| OBJETIVO Obtener la información en formato EDI proveniente de otras compañías. | | |
| FRECUENCIA Diario. Tantas veces como se reciba información de otras compañías. | | |
| OPERACIONES BASICAS <ul style="list-style-type: none"> ● Obtener la información a traducir de los archivos EDI recibidos. | | |
| ENTIDADES ACCESADAS TEAEDI. Archivos EDI. | | |
| ENTRADAS | SALIDAS | |

DESCRIPCION DE PROCESO

| | | |
|---|---|---------------------------|
| SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS. | | FECHA 04/DIC/90 |
| PROCESO 3.0 Traducción de formatos de la compañía | | |
| OBJETIVO Traducir la información del formato de la compañía a los estándares EDI correspondientes. | | |
| FRECUENCIA Diario. Tantas veces como se requiera enviar información a otras compañías. | | |
| OPERACIONES BASICAS | | |
| <ul style="list-style-type: none"> ● Identificar el formato a utilizar de la compañía. ● Identificar los datos del compañero EDI. ● Crear la transacción con los datos de la compañía. ● Agregarle la información del compañero EDI. ● Almacenar la transacción en la base de datos de transacciones. ● Almacenar la respuesta en el archivo de control de transacciones. | | |
| ENTIDADES ACCESADAS | | |
| TEFCIA Formatos de la compañía | TEBDEDI Base de Datos EDI | |
| TECEDI Archivos de Compañeros EDI | TEBDCT Base de Datos de Control de transmisión | |
| ENTRADAS | SALIDAS | |

DESCRIPCION DE PROCESO

| | | |
|---|----------------------------------|---------------------------|
| SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS. | | FECHA 04/DIC/90 |
| PROCESO 4.0 Traducción de formatos EDI | | |
| OBJETIVO Traducir la información de los formatos EDI a los formatos utilizados por la compañía. | | |
| FRECUENCIA Diario. Tantas veces como se reciba información de otras compañías. | | |
| OPERACIONES BASICAS <ul style="list-style-type: none"> ● Analizar el Grupo Funcional de la Transacción e Identificar los datos del compañero EDI. ● Traducir la transacción y almacenar el resultado en la base de datos de la compañía. ● Generar la transacción de acuse de recibo de la Información. | | |
| ENTIDADES ACCESADAS | | |
| TEBDCIA Base de Datos de la compañía | TEBDEDI Base de Datos EDI | |
| TECEDI Compañeros EDI | | |
| ENTRADAS | SALIDAS | |

2.11 DESCRIPCION DE ENTRADAS/SALIDAS

La descripción de entradas/salidas es la definición de la Interfaz que tiene con el sistema.

Las entradas y salidas son información que maneja el usuario (documentos, datos de pantalla, etc.). NO EL MEDIO POR EL CUAL SE RELACIONE CON EL SISTEMA.

Las entradas/salidas constan de los siguientes puntos:

Nombre de la entrada/salida

Objetivo

Funciones involucradas.- Procesos que utilizan la información de la entrada/salida.

Volumen y frecuencia.- Cuanta de ésta información se utilizará y con que frecuencia.

Enumeración de los elementos datos que la componen.

Observaciones.- Consideraciones a tomar en los elementos datos.

En las siguientes páginas se muestran ejemplos del formato utilizado para representar las entradas y salidas.

DESCRIPCION DE ENTRADA-SALIDA

| | |
|---|----------------------------|
| SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS. | FECHA 04/DIC/90 |
| ENTRADA-SALIDA TEFOR410 Factura y detalles de carga | USUARIO Operador |
| OBJETIVO Proporcionar información sobre los servicios de transportación efectuados y el monto a pagar derivado de estos. | |
| FUNCION INVOLUCRADA Recepción de Información de la compañía. Recepción de información EDI. | |
| VOLUMEN Y FRECUENCIA Diario.Tantas veces como transacciones de facturas se realicen. | |
| DATOS QUE CONTIENE Y CLASIFICACION ST Encabezado de transacción B3 Inicio de segmento para factura de transporte B3B Inicio de segmento para factura de transporte (ferrocarril) C4 Monto alternado pagado N9 Número de referencia N7 Detalles de equipo VC Control del vehículo motor G4 Segmento de identificación de escala M7 Números de sellos N5 Carro ordenado IC Equipo de chasis intermodal N8 Referencia del número de guía F9 Estación origen D9 Estación destino F1 Nombre del remitente F2 Dirección del remitente F4 Ciudad del remitente D1 Nombre del consignatario D2 Dirección del consignatario D4 Ciudad del consignatario U1 Nombre del último consignatario U2 Dirección del último consignatario U4 Ciudad del último consignatario U5 Nombre del origen anterior U6 Dirección del origen anterior | OBSERVACIONES |

DESCRIPCION DE ENTRADA-SALIDA

| | |
|--|------------------------------------|
| <p>SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS.</p> | <p>FECHA 04/DIC/90</p> |
| <p>ENTRADA-SALIDA TEFOR410 Factura y detalles de carga</p> | <p>USUARIO Operador</p> |
| <p>OBJETIVO Proporcionar Información sobre los servicios de transportación efectuados y el monto a pagar derivado de estos.</p> | |
| <p>FUNCION INVOLUCRADA Recepción de Información de la compañía. Recepción de Información EDI.</p> | |
| <p>VOLUMEN Y FRECUENCIA Diario. Tantos veces como transacciones de facturas se realicen.</p> | |
| <p>DATOS QUE CONTIENE Y CLASIFICACION</p> <ul style="list-style-type: none"> U9 Ciudad del origen anterior F5 Tercer remitente F6 Dirección del tercer remitente F7 Ciudad del tercer remitente D5 Tercer consignatario D6 Dirección del tercer consignatario D7 Ciudad del tercer consignatario S1 Nombre de Parada S2 Dirección de Parada S9 Estación de Parada R2 Información de ruta L5 Descripción, marcas y números L0 Artículo de línea/cantidad y peso L1 Tarifa y cargos L7 Referencias tarifarias T1 Origen del tránsito dentro de la frontera T2 Guía del tránsito dentro de la frontera T3 Ruta del tránsito dentro de la frontera T6 Cuotas del tránsito dentro de la frontera T8 Datos de tránsito en formato libre L3 Peso total y cargos. SE Terminación de transacción | <p>OBSERVACIONES</p> |

2.12. DIAGRAMA DE ESTRUCTURA DE LA INFORMACIÓN

(Diagrama Lógico de la Base de DATOS)

Los pasos a seguir para identificar el Diagrama de la Estructura de la Información son los siguientes:

- Identificar las Entidades de Información
- Determinar las relaciones entre las entidades de información
- Definir las Identidades de las entidades de Información
- Listar los elementos

Para identificar las entidades de la información se deben analizar las descripciones de las entradas y salidas de los procesos, así como el diagrama conceptual de usuario, además de aplicar segmentación y agrupación a los datos.

En la identificación de las relaciones entre entidades se establece una unión entre 2 entidades de información, dichas relaciones deberán tener un cuantificador.

La definición de la identidad (llave) garantiza que cada ocurrencia de la entidad exista en forma única. Estás ayudan a establecer las relaciones de las entidades de información.

En la lista de los elementos se deben identificar los datos que provienen de las entradas y aquellos de las salidas; así como las interrelaciones entre funciones.

En la figura 2.13 se muestra el diagrama de la estructura de información de las entidades utilizadas para la generación de una transacción:

Diagramas Lógicos de la Base de Datos

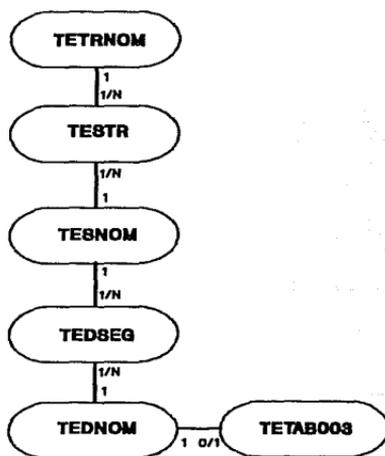


Figura 2.13 Diagrama de Estructura de información (generación de transacción)

Sistema Traductor de Intercambio Electrónico de Datos

Una transacción (TETRNOM) tiene asociados varios segmentos, esta relación está almacenada en (transacciones-segmentos) (TESTR).

Cada uno de los registros de transacciones-segmentos (TETRNOM) tiene asociada una descripción de segmento (TESNOM).

Un segmento (TESNOM) tiene asociado varios datos, esta relación está almacenada en la entidad datos-segmentos (TEDSEG).

Cada uno de los registros de segmento-dato (TEDSEG) tiene asociado una descripción de los datos (TEDNOM).

Un dato puede o no tener un catálogo asociado con la información válida para dicho dato, el cual se encuentra identificada en el catálogo de elementos datos (TETAB003).

Nota: En las descripciones de los datos y segmentos se incluyen las características específicas del segmento o dato, aquéllas que no sufren ninguna modificación independientemente de la transacción en la que sean utilizados.

En la figura 2.14 se muestra el diagrama de la estructura de Información de las entidades utilizadas para el manejo de la transacción:

Una transacción enviada (TEENVTR) tiene asociado un registro de control de conocimiento de transacciones (TECONTR) y tiene asociado únicamente un compañero EDI (TECEDI).

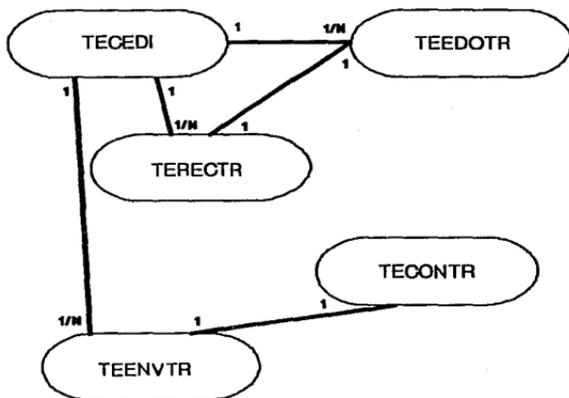


Figura 2.14 Diagrama de Estructura de Información (manejo de transacción)

Un compañero EDI (TECEDI) puede tener una o varias transacciones enviadas (TEENVTR) y transacciones recibidas (TERECTR), así como uno o más estados (aceptaciones/rechazos) de las transacciones enviadas (TEEDOTR).

Una transacción recibida (TERECTR) tiene uno o varios estados de las transacciones (TEEDOTR) y un compañero EDI (TECEDI).

2.13 DICCIONARIO DE DATOS

El diccionario de datos contiene las definiciones de todos los datos mencionados en el Diagrama de Flujo de Datos, en una especificación del proceso y en el propio diccionario de datos. Los datos compuestos (datos que pueden ser además divididos) se definen en términos de sus componentes; los datos elementales (datos que no pueden ser divididos) se definen en términos del significado de cada uno de los valores que pueden asumir. Por tanto, el diccionario de datos está compuesto de definiciones de flujo de datos, archivos (datos almacenados) y datos usados en los procesos (transformaciones).

La notación de un diccionario de datos, facilita al analista la representación de los datos compuestos en una de las tres formas fundamentales en que puede ser construido: 1) como una secuencia de elementos de datos; 2) como una selección entre un conjunto de elementos de datos, o 3) como una agrupación repetida de elementos de datos. Cada entrada de un elemento de datos que se representa como parte de una secuencia, selección o repetición, puede a su vez ser otro elemento de datos compuestos, el cual necesita un refinamiento posterior dentro del diccionario.

| Construcción de datos | Notación | Significado |
|-----------------------|----------|------------------------------------|
| | = | Está compuesto de y |
| Secuencia | + | Uno u otro |
| Selección | { } | n repeticiones de Datos opcionales |
| Repetición | { }n | Aparece si aparecen otros |
| Condición | () | |
| | * | |

En la siguiente página se muestran algunos ejemplos del formato utilizado para la generación del diccionario de datos.

DICCIONARIO DE DATOS

| <p>SISTEMA TE TRADUCTOR PARA INTERCAMBIO ELECTRONICO DE DATOS.</p> | <p>FECHA 04/DIC/90</p> |
|---|-----------------------------------|
| <p>DESCRIPCION DEL DATO</p> | |
| <p>AMT Importe monetario = [Código de monto + Monto monetario]</p> <p>BCH Inicio de segmento para cambio de orden de compra = [Propósito del conjunto de transferencia CD + Tipo de orden de compra + Número de orden de compra + (Número de liberación) + (Secuencia de orden de carga) + Fecha de la orden de compra + (Número de referencia de requerimiento) + (Número de contrato) + (Número de referencia) + (Fecha de aceptación) + (Fecha de requerimiento de carga de la orden de compra)</p> <p>BEG Inicio de segmento para orden de compra = Propósito del conjunto de transferencia CD + Tipo de orden de compra + Número de orden de compra + (Número de liberación) + Fecha de la orden de compra + (Número de contrato) + (Tipo de aceptación)</p> <p>BGF Inicio segmento para transferencia de información de archivos = (Identificador de transacción) + Número de referencia de calificador + Número de referencia</p> | |

CAPITULO 3

diseño del sistema

3.1. AMBITO

3.1.1 OBJETIVOS DEL SISTEMA.

El objetivo del trabajo de tesis es desarrollar un sistema de traducción de formatos EDI, que funcione como un preproceso ("front end") para aplicaciones que contemple 10 formatos públicos que serán utilizados en Ferrocarriles Nacionales de México.

3.1.2 HARDWARE, SOFTWARE E INTERFACES HUMANAS.

Uno de los aspectos más importantes a considerar durante la planificación del desarrollo de sistemas es la estimación de los recursos requeridos. Como base de estos recursos se tienen al *hardware* y al *software*, considerados como el soporte del esfuerzo de desarrollo. En un nivel más alto se encuentra el recurso humano, el cual está altamente interrelacionado con los recursos anteriores.

A continuación se muestran los recursos de *hardware* y *software* y las características que el grupo de trabajo observó para cada uno de ellos.

Hardware :

Mainframes.

- Ventajas :
- Alta capacidad de procesamiento.
 - Gran capacidad de almacenamiento.
 - Multiprocesamiento.
 - Mayor cobertura.

- Desventajas:
- Falta de conocimiento global.
 - Alto grado de especialización.

- Baja disponibilidad de equipo para los integrantes del grupo de trabajo.
- Alto costo.

Minicomputadoras.

- Ventajas :
- Mediana capacidad de procesamiento.
 - Mediana capacidad de almacenamiento.
 - Multiprocesamiento.
 - Cobertura medía.

- Desventajas:
- Falta de conocimiento global.
 - Cierto grado de especialización.
 - Baja disponibilidad de equipo para los integrantes del grupo de trabajo.
 - Mediano costo.

Microcomputadoras.

- Ventajas :
- Alta disponibilidad.
 - Bajo grado de especialización.
 - Bajo costo de recursos.
 - Gran variedad de productos de *software* utilizables.

- Desventajas:
- Uniprocesamiento.
 - Baja capacidad almacenamiento

Software :

Lenguajes de 4a. Generación.

- Ventajas :
- Menor tiempo para el desarrollo.
 - Sencillez para la generación de pantallas y reportes.

Sistema Traductor de Intercambio Electrónico de Datos

- Desventajas:**
- Métodos óptimos de acceso a la Información.
 - Flexibilidad para cambios..
 - Se requiere gran capacidad de procesamiento.
 - Alto costo.
 - Transportabilidad limitada
 - Manejo limitado de cadenas de caracteres de longitud variable.

Lenguajes de 3a. Generación.

- Ventajas :**
- Alta transportabilidad.
 - Consume menos recursos.
 - Amplio conocimiento de los Integrantes del grupo.
- Desventajas:**
- Mayor dificultad en el manejo de archivos.
 - Mayor esfuerzo del programador para el desarrollo.
 - Mayor esfuerzo para mantenimiento

Como se podrá observar, la tendencia en cuanto a *Hardware* es el uso de microcomputadoras, ya que presentan la capacidad necesaria para soportar al sistema, y además se cuenta con una gran variedad de *software* para el desarrollo.

En lo que se refiere al *software*, es posible trabajar con varios de ellos, sin necesidad de limitarse a uno solamente. Con esto, es posible facilitar el desarrollo del sistema, debido a que cada uno de ellos tiene una función específica.

En lo que respecta a las Interfaces humanas, los usuarios del sistema propuesto, deberán tener un conocimiento básico sobre el manejo de una computadora personal, como es el manejar comandos básicos del sistema

operativo MSDOS, para fines de respaldo del sistema o entrada al mismo. Se requiere al menos de un operador en cada uno de los extremos de la transmisión de la información, con el fin de que reciban la Información proveniente de otras compañías o puedan generar transacciones EDI, para ser enviadas a algún compañero de negocios. El nivel requerido para el operador es el de una persona con conocimientos técnicos básicos de una computadora personal.

3.1.3 PRINCIPALES FUNCIONES DEL SOFTWARE.

El *software* deberá ser capaz de :

1. Recibir las transacciones provenientes de los compañeros EDI, verificando identificador y clave de acceso (password) para evitar transacciones erróneas
2. Traducir las transacciones recibidas a formatos de la compañía, validando que la información recibida concuerde con las especificaciones de datos de los formatos EDI
3. Depositar la información, bajo el formato adecuado, en la Base de Datos de la compañía
4. Generar los reportes y mensajes necesarios que indiquen los errores encontrados durante los procesos de recepción y traducción

además de :

1. Identificar la transacción que se desea generar, así como el formato de la compañía correspondiente
2. Identificar al compañero EDI con el cuál se efectúa la transacción
3. Extraer información propia de la compañía, bajo el formato adecuado, y traducirla al formato EDI correspondiente, validando que la información concuerde con las especificaciones de datos de los formatos EDI

diseño del sistema

4. Agrupar las transacciones comunes (formar grupos funcionales) y enviarlas al compañero EDI, así como almacenarlas para tener un control de transacciones
5. Generar adecuadamente aquellas transacciones cuya respuesta, después de haber sido enviadas, indique que fueron recibidas con error

3.1.4 BASE DE DATOS DEFINIDA EXTERNAMENTE.

Base de Datos de la compañía : esta base de datos esta definida externamente, ya que cada compañía definirá la base de datos que mejor se adecue a sus necesidades. El sistema no tiene control sobre el formato que dicha base presente, pero si debe ser capaz de basarse en su formato para generar transacciones o recibir información de transacciones provenientes de los compañeros EDI.

3.1.5 PRINCIPALES LIGADURAS Y LIMITACIONES DEL DISEÑO.

El sistema estará ligado al *software* utilizado. En caso de transportarse a equipos con poca o nula compatibilidad, éstos deberán contar con la versión de *software* propio del equipo (ej. Turbo Pascal para PC's, PASCAL para equipos DIGITAL), siendo necesario modificar aquellas funciones que no sean soportadas por el nuevo *software* a utilizar. El sistema sólo podrá trabajar con aquellos formatos previamente definidos.

3.2. DOCUMENTOS DE REFERENCIA.

3.2.1 DESCRIPCIÓN DEL SOFTWARE EXISTENTE

Existen seis tipos de vendedores de formatos EDI en los países desarrollados: Compañías de *software* de traducción y aplicación, Compañías de redes de valor agregado,

Bancos, Compañías de computadoras, y Consultores EDI. Estas compañías venden diferentes tipos de *software* de traducción de formatos EDI y de comunicaciones.

Existen tres tipos de *software* de traducción:

- los que funcionan sólo para recibir información, es decir sólo la imprimen;
- aquellos que envían y reciben pero como un "front end" para la aplicación; y
- los que pueden realizar comunicación de aplicación a aplicación.

El *software* de comunicaciones es el que permite enviar los datos utilizando la red de comunicaciones. Tiene el control de los protocolos a utilizar en la conexión con otras computadoras.

Existen catálogos sobre vendedores de productos EDI, en los cuales se pueden encontrar consultores, bancos, servicios de red, publicaciones, pasando por educación y vendedores de *software*.

Para dar una idea de la importancia de EDI se enlistan a continuación algunas de las compañías más importantes que ofrecen productos EDI:

- General Electric Information Services.
- First Interstate Bank.
- The First National Bank of Chicago.
- AT&T.
- Control Data Corp. REDINET Services.
- Sears Communication Company.
- Western Union Corporation.
- International Business Machines (IBM).
- Digital Equipment Corporation (DEC).
- McDonnell Douglas.

El *software* se distribuye en los Estados Unidos, Canadá, Europa, Reino Unido, Unión Soviética, África, Japón, Nueva Zelanda, Australia y Suroeste de Asia.

Sistema Traductor de Intercambio Electrónico de Datos

En México la empresa IBM de México anunció en enero de 1991 un software de traducción EDI para diferentes rangos de equipo, y es la única empresa que distribuye EDI en el país. El software se distribuye para los diferentes rangos de equipo de cómputo, es decir, micros, minis y mainframes.

Los precios varían de acuerdo al tipo de traductor o de servicio, y dependiendo de la calidad del mismo. A continuación se presenta una estimación del costo del mismo:

| Software de Traducción EDI | | |
|----------------------------|------------|----------|
| | mínimo | máximo |
| MICRO | \$ 700 - | \$ 5,500 |
| MINI | \$ 8,000 - | \$11,000 |
| IBM AS-400 | \$10,000 - | \$25,000 |
| MAINFRAME | \$12,000 - | \$30,000 |

3.2.2 DESCRIPCIÓN DE SISTEMAS EXISTENTES

Sistema actual para el pago del perdiem y el millaje

El principal objetivo de la contabilidad de carros de ferrocarril es el determinar la compensación debida al uso de su equipo de material rodante. Esta compensación toma la forma de honorarios de arrendamiento para cada hora de uso por el ferrocarril no propietario, y/o una cuota de uso basada en el número de millas recorridas.

Siempre en un esfuerzo para fomentar la compra, adquisición y una utilización eficiente de los vagones de carga, el objetivo ha sido una compensación equitativa. Por consiguiente, varios elementos afectan los costos y cuotas de la Renta de Carros (CAR HIRE) como son costos de capital, depreciación, mantenimiento de carros, impuestos,

factores de disponibilidad de transportación (días de servicio y millas acumuladas) y ocasionalmente elementos Incentivos tienen que ser tomados en consideración.

En años pasados, la contabilidad para los carros ha sido difícil de manejar y administrativamente cara. La implantación del Sistema de Intercambio de Datos del Car Hire en la Asociación Americana de Ferrocarriles (AAR *Association of American Railroads*) ha reducido enormemente estos costos por la automatización, estandarización y centralización de la colección de datos para su contabilidad y la distribución de procesos. Costos menores fueron observados en las funciones de captura de datos y auditoría, y estas reducciones de costos se ha hecho más impresionantes a medida que el número de participantes ha crecido.

En el presente, las cintas de intercambio de datos representan alrededor de 1,800 marcas de ferrocarriles, que llegan a la AAR para su procesamiento cada mes. Unos 2.5 millones de registros que involucran cerca de 250 millones de dólares en concesiones son leídos, sorteados, fusionados y escritos como salida para 103 socios-subscriptores.

En un tiempo el uso de los carros de ferrocarril fue compensado en forma tan sencilla como la cuota diaria (llamado perdiem) y los propietarios de carros privados fueron compensados en base a millas recorridas (usualmente en cuotas basadas según el tipo de equipo). No se tomaba ninguna condición acerca de la edad o valor del carro, aunque los niveles de compensación ocasionalmente se incrementaban porque los costos subían. Gradualmente, se reconoció que esto daba como resultado muchas desigualdades (por ejemplo, un carro muy viejo tenía la misma compensación que una nuevo). Eventualmente, la edad y el valor del carro fueron tomados en consideración.

diseño del sistema

Actualmente el pago de la renta de carros se envía, después de un proceso semiautomatizado, a cada una de las compañías de ferrocarriles dueñas de los carros que se intenan en la república mexicana. Los periodos de pago son mensuales, y se generan alrededor de 500 páginas, las cuales son distribuidas por medio de un servicio de mensajería. El programa de intercambio de la AAR permite el envío de una sola cinta magnética con la información perteneciente a todos los carros que se deben pagar, y este sistema se encarga de la distribución de la misma, con lo cual se ahorra mucho tiempo y esfuerzo.

La AAR cambió sus formatos de intercambio de datos de renta de carros a formatos de intercambio electrónico de datos (EDI), con el fin de solventar las carencias y deficiencias de los registros de longitud fija. Por lo anterior, los Ferrocarriles Nacionales de México deben desarrollar un sistema que maneje los estándares EDI para poder seguir intercambiando información de manera electrónica con los ferrocarriles americanos.

Sistema de facturación computarizada (SIFAC).

El Sistema de Facturación Computarizada tiene como función generar y almacenar toda la información referente a las facturas, originadas por los servicios prestados por los Ferrocarriles Nacionales de México a los individuos que así lo soliciten.

Básicamente maneja dos documentos :

1. Conocimiento
2. Cuenta de Gastos

El primero es el documento que se genera en la estación origen y su función es hacer el conocimiento de una solicitud de transporte, aunque este documento presenta una serie de copias, cada una de éstas posee un nombre propio y

tiene un fin diferente, pese a que cuentan con la misma información. Además, el documento en sí podría ser una especie de factura, ya que en este punto se hace un adelanto por el servicio solicitado, a cambio se entrega un comprobante de pago.

Inicialmente el solicitante de servicio debe llenar un requerimiento de servicio, a partir de éste se capturan todos los datos referentes al servicio solicitado, los cuales son almacenados en la base de datos. Posteriormente, y a petición del operador de la computadora, se genera el reporte correspondiente. Una de las copias de este documento es entregado al cliente, mientras otra copia (Copia Guía) se entrega a la persona encargada del control de la mercancía durante su traslado.

El documento de Cuenta de Gastos, considerado así como factura, se genera en la estación destino cuando los Ferrocarriles Nacionales de México hacen entrega de la mercancía transportada al remitente, en este punto se hace el cobro de la cantidad restante a pagar por el monto total del servicio. En este caso, como en el anterior, el documento cuenta con varias copias, las cuales tienen nombre propio y funciones diferentes.

Para el cálculo del monto total se basan en lo reportado en la Copia Guía, si durante el trayecto a la estación destino se presentaron algunos percances ocasionados por la mercancía transportada, o si fue necesario darles mantenimiento, o si requirió servicios especiales, o algún servicio adicional, todos estos aspectos son registrados en la copia e implican un monto adicional de pago. Los datos son registrados en la computadora de la estación destino para generar la Cuenta de Gastos, cuyo monto a pagar deberá ser cubierto por el remitente.

Sistema Traductor de Intercambio Electrónico de Datos

Aviso de adeudos.

El documento de Aviso de Adeudo se origina cuando un cliente no ha cubierto el monto a pagar por los servicios prestados por los Ferrocarriles Nacionales de México. Muchos de estos clientes cuentan con crédito por parte de los Ferrocarriles; sin embargo, cuando se llegan a atrasar en sus pagos, la compañía considera necesaria enviarles un aviso indicándoles este hecho.

3.3. DESCRIPCIÓN DEL DISEÑO.

3.3.1 METODOLOGÍA DE DISEÑO

El método de diseño utilizado para la realización de este sistema es el orientado al flujo de datos. El diseño se ha descrito como un proceso en el que las representaciones de la estructura de datos, estructura de programa y procedimiento, se sintetizan a partir de los requerimientos de la información. El diseño es una actividad referida a la toma de decisiones importantes, frecuentemente de la naturaleza estructural. Comparte con la programación los aspectos referentes a la abstracción en la representación de la información y en las secuencias de procesamiento, pero el nivel de detalle es muy diferente en ambos casos. El diseño construye representaciones coherentes y bien planificadas de programas, concentrándose en las interrelaciones de las partes a un nivel más alto y en las operaciones lógicas implicadas en los niveles inferiores.

El diseño orientado al flujo de datos tiene sus orígenes en los primeros conceptos de diseño que aventuraron la modularidad, diseño descendente y programación estructurada. Sin embargo, el enfoque de diseño orientado al flujo de datos amplió esas técnicas procedimentales, integrando explícitamente el flujo de la información en el proceso de diseño.

El diseño orientado al flujo de datos permite una cómoda transición de las representaciones de la información (el diagrama de flujo de datos DFD) a una descripción de diseño de la estructura del programa. La transición desde el flujo de la información a la estructura, se realiza como parte de un proceso de cinco pasos: 1) se establece el tipo de flujo de información; 2) se indican los límites del flujo; 3) el DFD se convierte en la estructura del programa; 4) se define la jerarquía de control mediante factorización, y 5) se refina la estructura resultante usando medidas y heurísticas de diseño.

Existen dos tipos de flujo de información, los cuales conducen los métodos de conversión mencionados en el paso 3. El flujo de transformación y el de transacción. El primero se representa por una estructura de flujo en la cual se percibe una "entrada" de información al sistema, mediante caminos que transforman los datos externos en una forma interna y se identifica como flujo de llegada. En el núcleo del *software* ocurre una transición. Los datos de llegada se pasan a través del centro de transformación y comienzan a moverse a lo largo de caminos que conducen ahora a la "salida" del *software*. Los datos que se mueven por estos caminos se llaman flujo de salida. Cuando un segmento de un DFD exhibe estas características se presenta un flujo de transformación.

El flujo de transacción se caracteriza por datos que se mueven a lo largo de un camino de llegada que convierte información del mundo externo en una transacción. La transacción es evaluada, y basándose en su valor, el flujo se inicia por uno de los caminos de acción. El centro de flujo de información desde el que emanan muchos caminos de acción se llama un centro de transacción.

Es importante resaltar que en un DFD para un gran sistema pueden presentarse los dos tipos de flujos, de transformación y de transacción.

3.3.1.1 ACOPLAMIENTO Y COHESION
ACOPLAMIENTO

El acoplamiento es una medida de la interconexión entre módulos en una estructura de programa. El acoplamiento depende de la complejidad de la interfaz entre módulos, el punto en el que se hace una entrada o referencia a un módulo y los datos que pasan a través de la interfaz.

En el diseño de *software* se busca el acoplamiento más bajo posible. Una conectividad sencilla entre módulos da como resultado un *software* que es más fácil de comprender y menos propenso al "efecto onda" causado cuando los errores ocurren en una posición y se propagan a lo largo del sistema.

A niveles moderados, el acoplamiento se caracteriza por el paso de control entre módulos. Niveles relativamente altos de acoplamiento se producen cuando los módulos están ligados a un entorno externo al *software*; por ejemplo, la entrada/salida acopla un módulo a dispositivos, formatos y protocolos de comunicación específicos. El acoplamiento externo es esencial, pero debe limitarse a un pequeño número de módulos dentro de una estructura. Se presenta también un alto acoplamiento cuando varios módulos referencian a una área de datos global, este tipo de acoplamiento se llama común.

El acoplamiento por contenido es el peor, y ocurre cuando un módulo hace uso de información de control o de datos mantenidos dentro de los límites de otro módulo. Secundariamente, este tipo de acoplamiento ocurre cuando se bifurca a la mitad de un módulo, este modo de acoplamiento puede y debe evitarse.

COHESION

La cohesión es una medida de que tan fuerte es la asociación de los elementos dentro de un módulo.

Un elemento puede ser una instrucción, un grupo de instrucciones o una llamada a otro módulo.

Lo deseable es tener módulos fuertes y altamente cohesivos, módulos cuyos elementos están genuinamente relacionados.

La cohesión es una medida secundaria para saber que tan bien está particionado el sistema.

Los niveles de cohesión son:

| | |
|----------------|-------|
| Funcional | MEJOR |
| Secuencial | |
| Comunicacional | |
| Procedural | |
| Temporal | |
| Lógico | |
| Colocidental | PEOR |

Un módulo funcionalmente cohesivo es aquel en que todos sus elementos contribuyen a una y sólo una tarea completa.

Un módulo secuencialmente cohesivo es uno cuyos elementos están involucrados en tareas donde los datos de salida de una de ellas sirven como datos de entrada para la siguiente. Un módulo secuencialmente cohesivo usualmente muestra buen acoplamiento.

Un módulo comunicacionalmente cohesivo es aquel cuyos elementos contribuyen a diferentes tareas, pero cada una de éstas tareas refieren a la misma entrada o salida de

Sistema Traductor de Intercambio Electrónico de Datos

parámetros. Al realizar una división funcional de módulos con este tipo de función, se simplifica el acoplamiento y se incrementa la cohesión.

Un módulo proceduralmente cohesivo es uno en el cual el control pasa de un elemento al siguiente. Sin embargo, los datos no necesariamente fluyen de un elemento al siguiente. El más grande problema con un módulo procedural es que se pasan resultados internos parciales, banderas, *switches*, etc.

Un módulo temporalmente cohesivo es aquel donde sus elementos están relacionados en el tiempo. Sin embargo, usualmente, estos elementos en realidad pertenecen a diferentes funciones. Contiene elementos cuya única relación es que en la implantación de un particular flujo de datos ellos se mueven al mismo tiempo. Pero la inicialización o terminación, etc., de muchas diferentes funciones es a menudo contenida en un módulo temporal, el cual:

- rompe la regla acerca de mantener las cosas que no están relacionadas aparte, y
- proporciona al programador una oportunidad para compartir código, etc.

Un módulo lógicamente cohesivo es uno cuyos elementos "parecen" que están involucrados en tareas de la misma categoría general. Este tipo de módulos causan problemas de mantenimiento, porque las razones usuales para el programador de poner diferentes funciones en un mismo módulo son:

- Compartir *buffers*, o
- Compartir código, o hasta
- Compartir constantes o parámetros de entrada/salida

Un módulo coincidentalmente cohesivo tiene elementos con relaciones que no tienen significado para los demás. Esto usualmente realiza diferentes funciones que no tienen relación entre sí para diferentes jefes.

3.3.1.2 REVISIÓN DEL FLUJO DE DATOS.

Procesos de Generación de Transacción

Al aplicar las técnicas de diseño estructurado (orientado al flujo de datos), al diagrama de segundo nivel del proceso de Traducción a Información EDI (ver fig. 2.8, capítulo 2), se obtuvieron los diagramas de flujo de datos de las figuras 3.1, 3.2 y 3.3.

Al aplicar el análisis de cohesión entre módulos, al diagrama de flujo de datos obtenido en el capítulo 2, se observó que los procesos englobaban muchas funciones, por lo que fue necesario generar nuevos procesos con una función específica cada uno de ellos.

A continuación se indican los cambios que sufrieron cada uno de los procesos contemplados en el diagrama de flujo inicial:

Identifica formato de la Compañía (2.6).- Este no sufrió ninguna modificación y su objetivo es identificar el formato de la compañía asociado a la transacción a generar.

Identifica compañero EDI (2.2).- Sin modificaciones, su finalidad es obtener la información del compañero al cual se le enviará la transacción.

Creación de Transacción.- Se dividió en los siguientes procesos:

Obten segmento dato (2.3).- Este proceso tiene como función principal obtener los segmentos de la transacción así como los datos asociados al mismo (segmento-dato).

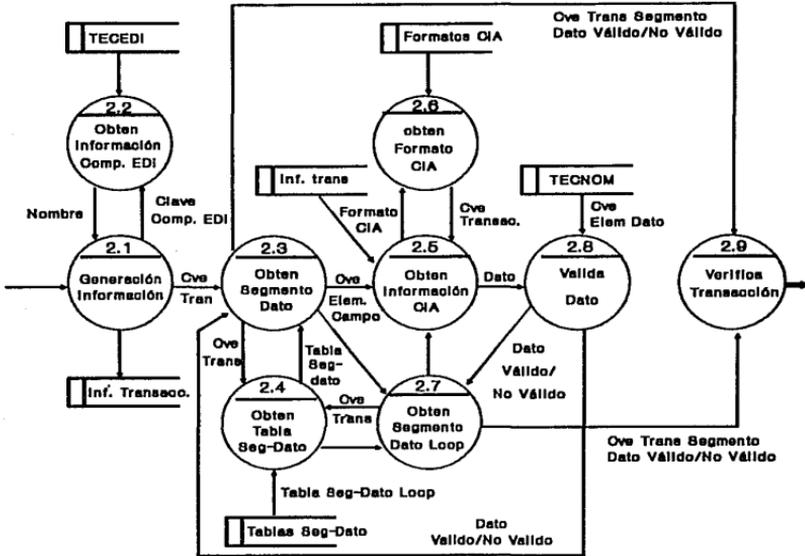


Figura 3.1. Diagrama de flujo de datos de generación de transacción (general)

Obten segmento dato *loop* (2.7).- El objetivo de este proceso es el de obtener los segmentos y datos que presentan iteración.

Obten tabla segmento dato (2.4).- El proceso accesa las tablas segmentos dato y determina cual corresponde a la transacción que se está realizando.

Obten Información de la compañía (2.5).- El objetivo de éste es obtener la Información de la compañía asociada al segmento-dato de la transacción.

Valida dato (2.8).- Valida que la información de la compañía cumpla con las características del segmento-dato.

Sistema Traductor de Intercambio Electrónico de Datos

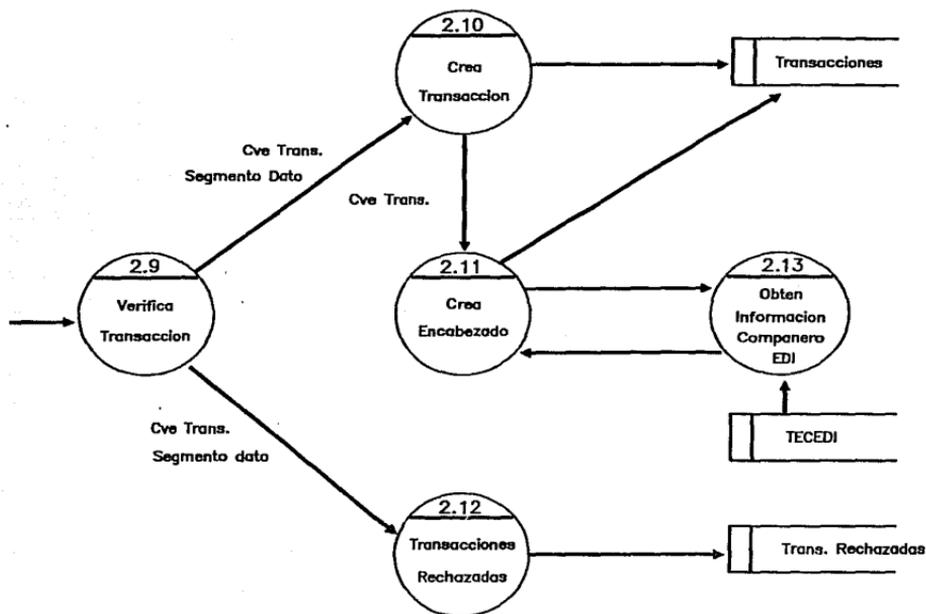


Figura 3.2. Diagrama de flujo de envío o rechazo de transacción.

Escribe en EDI.- Se dividió en los siguientes procesos:

Verifica transacción (2.9).- Este tiene como objetivo aceptar o rechazar la transacción dependiendo de las validaciones hechas a los segmentos-datos.

Crea transacción (2.10).- El objetivo del proceso se encarga de crear el archivo con las transacciones válidas.

Crea encabezado (2.11).- Genera el encabezado de transacción, en el cual se indica la identificación de la transacción y del compañero EDI.

Transacciones rechazadas (2.12).- Almacena aquellas transacciones que tuvieron alguna información errónea.

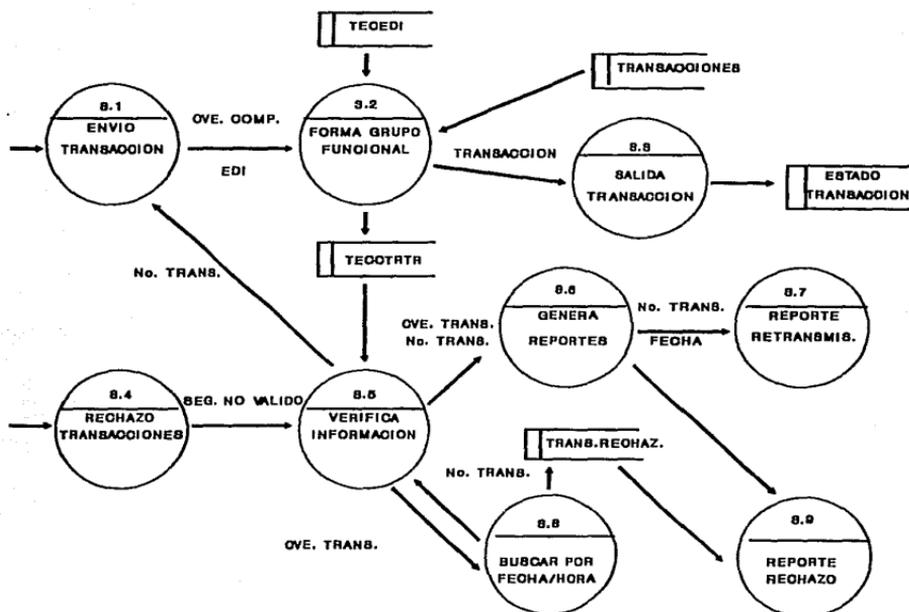


Figura 3.3. Diagrama de flujo de datos de generación de transacciones.

Adicionalmente a la división de funciones, hubo necesidad de generar nuevos procesos para cubrir nuevas funciones no contempladas con anterioridad. Por ejemplo, en el diagrama de la figura 3.3, se muestra el proceso de rechazo de las transacciones:

Rechazo de transacciones (3.4).- Este proceso toma aquellas transacciones que fueron recibidas como respuesta a una transacción pero que tuvieron algún problema en la transmisión de la información.

Verifica Información (3.5).- El objetivo de éste es verificar que la información recibida de la transacción rechazada exista como transacción enviada. Además, toma la información para enviarla nuevamente.

Sistema Traductor de Intercambio Electrónico de Datos

Genera reportes (3.6).- En éste se determina con que frecuencia se obtienen los reportes informativos, esto es de transacciones rechazadas o retransmitidas.

Algunos procesos son utilizados por otros, sin que exista una dependencia entre ambos, ya que cada uno tiene una función específica. Como ejemplo se tiene el proceso valida dato que ya fue explicado anteriormente.

Procesos de Traducción de la Información

Los cambios más importantes que se llevaron a cabo entre el diagrama de flujo de datos para la traducción de información presentado en el capítulo 2 y el de la figura 3.4 son:

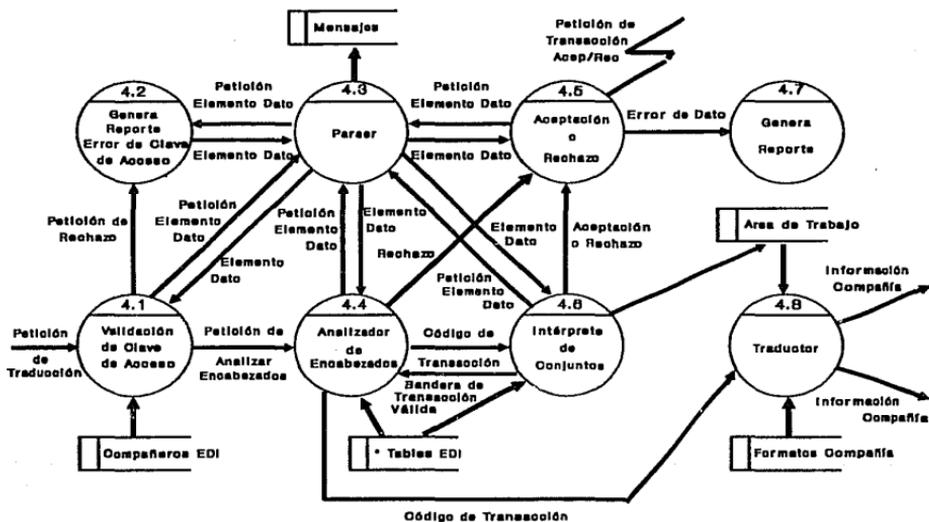


Figura 3.4. Diagrama de flujo de datos de traducción de transacciones.

El módulo 4.1 (**Validación de Clave de Acceso**) generaba una petición de transacción de rechazo, pero en la revisión se llegó a la conclusión de que un error en la clave de acceso supone problemas de seguridad, por lo que se tratará este asunto entre los responsables de la seguridad del Intercambio de Información. En consecuencia, se modificó este módulo par que genere petición de reporte de error de clave de acceso. Por lo anterior, en el nuevo diagrama aparece el módulo 4.2 (**Genera Reporte de Error de Clave de Acceso**), el cual utiliza el 4.3 (**Parser**), para que le proporcione los elementos datos del encabezado de la transacción que aparecerán en el reporte.

Anteriormente, el módulo 4.4 (**Analizador de Encabezados**) le transfería el control al proceso 4.5 (**Intérprete de Conjuntos**), pero todo el mensaje por parte de un compañero EDI está lleno de encabezados (control de transmisión, de grupo funcional y de transacción), por lo tanto el módulo indicado para llevar el control de las iteraciones necesarias para validar los datos es el 4.4, y no el 4.5.

Debido a que el módulo 4.4 es el que detecta un fin de transacción, es el indicado para generar la petición de traducción de la Información que fue validada, y envía el código de la transacción al módulo 4.8 (**Traductor**), el cual a su vez lee los datos del Area de Trabajo, y los coloca en los formatos de la compañía.

El módulo 4.2 (**Aceptación o Rechazo**) generaba una petición de escritura a **Escribe en EDI**, el cual generaba una transacción de aceptación o rechazo según el caso y la escribía en la base de datos EDI. Después de la revisión, se detectó la duplicidad de trabajo, ya que existen módulos de generación de transacciones EDI. Por tal motivo, se eliminó el módulo **Escribe en EDI** y se modificó el 4.2 para que

obtuviera los datos necesarios para la transacción de aceptación o rechazo con la ayuda del **Parser** y envíe una petición de transacción a los módulos de generación.

Se incluyó el módulo 4.7 (**Genera Reporte**) que produce estadísticas de las transacciones rechazadas o aceptadas, con el fin de identificar cuales y bajo que circunstancias se presentan los problemas.

3.3.2 ESTRUCTURA DEL PROGRAMA

3.3.2.1 DESCRIPCION DE LAS ESTRUCTURAS DEL PROGRAMA

La estructura del programa representa la organización, frecuentemente jerárquica, de los componentes del programa (módulos) e implica una jerarquía de control. Esta estructura es derivada del diagrama de flujo de datos.

Para representar una estructura de programa se utilizan muchas notaciones diferentes. La más común es un diagrama de árbol, frecuentemente llamado diagrama de estructura, y es la que se utilizó en el presente trabajo. Para facilitar la comprensión de las estructuras se definen a continuación algunas medidas y términos sencillos.

Refiriéndonos a la figura 3.5, los términos de profundidad y anchura dan una indicación del número de niveles de control y de la expansión global de control respectivamente. El abanico de salida es una medida del número de módulos que están directamente controlados por otros módulos. El abanico de entrada indica cuantos módulos controlan directamente a un módulo dado.

Las declaraciones de control entre los módulos se expresan de la siguiente forma: un módulo que controla a otro módulo se dice que es superior a él, e inversamente, un módulo controlado por otro se dice que es un subordinado del controlador.

La heurística del diseño está basada en los siguientes criterios:

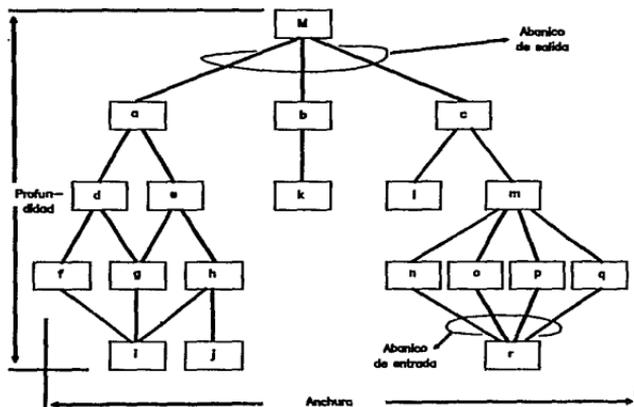


Figura 3.5. Diagrama de descripción de estructura de programa.

Todos los diagramas de flujo de datos de segundo nivel de este trabajo fueron transformados a estructuras de programas, y se explicaran más adelante.

3.3.2.2 DESCRIPCIÓN DE LAS HEURÍSTICAS DE DISEÑO

Para afinar la estructura de programa derivada del DFD, después de aplicar los criterios de flujo de transacción y transformación, se utilizan los criterio de heurística de diseño, los cuales se listan a continuación.

1.- Evaluar la estructura del programa preliminar para reducir el acoplamiento y aumentar la cohesión. Una vez que se ha desarrollado la estructura del programa, los módulos pueden expandirse o reducirse considerando siempre la independencia de los módulos.

2.- Intentar minimizar las estructuras con un abanico de salida ancho; fomentar los abanicos de entrada conforme incrementa la profundidad.

3.- Mantener el efecto de un módulo dentro del ámbito de control del mismo. El ámbito de efecto de un módulo se define como todos los módulos que quedan afectados por una decisión hecha en éste.

4.- Evaluar las interfaces de los módulos para reducir la complejidad, redundancia y mejora de la resistencia. Las interfaces deben diseñarse para pasar simplemente información y deben ser consistentes con la función de un módulo determinado.

5.- Definir módulos cuyas funciones sean predecibles, pero evitar que éstos sean demasiado restrictivos. Un módulo es predecible cuando puede ser tratado como una caja negra; esto es, los mismos datos externos se producirán independientemente de los detalles del procesamiento interno.

6- Buscar los módulos de entrada y salida única, evitando las conexiones complejas.

7.- Empaquetar el software basándose en las restricciones del diseño y requerimientos de transportabilidad. El empaquetado se refiere a las técnicas usadas para ensamblar el software para un entorno de procesamiento específico.

En dicho diagrama se presentan, inicialmente, características de transacción, lo cual, al aplicarle el análisis respectivo, derivó la presencia del módulo de Control de Transacciones, los flujos ligados a este último son los siguientes :

3.3.2.3 DESCRIPCIÓN DE LAS ESTRUCTURAS DE PROGRAMA FINALES.

En la figura 3.6 se muestra la estructura de programa derivada del diagrama de flujo de datos de la generación de la transacción (ver figura 3.1), este diagrama se obtiene al aplicar las técnicas de análisis de transacción y transformación, así como las heurísticas de diseño.

Flujo de Llegada .- La entrada al diagrama de estructura de programa esta compuesta por:

Generación de Información.
Obtención de la información del compañero EDI.

Estos módulos estan encargados de obtener la Información que será utilizada para crear la transacción EDI.

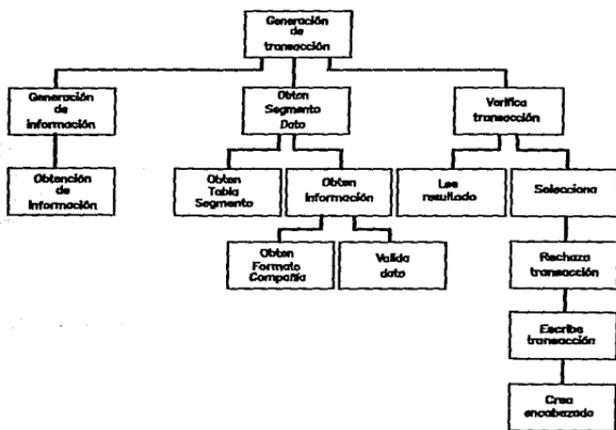


Figura 3.6. Estructura de programa derivada para la generación de transacciones.

Sistema Traductor de Intercambio Electrónico de Datos

Flujo de Transformación.- El proceso está formado por:

Obten segmento dato loop. Este presenta flujo de transformación, por lo cual esta constituido por:

Flujo de Llegada.- Obten Tabla segmento.
Flujo de Transformación.- Obten Información.

este último proceso, a su vez, presenta flujo de transformación, quedando integrado por:

Flujo de Llegada.- Obten formato de la Cía.
Flujo de Transformación.- Valida dato.

Todos estos módulos están encargados de procesar y validar la información.

Flujo de Salida.- La salida esta formada por:

Verifica transacción.

este módulo, derivado de aplicar el análisis de transacción, presenta los siguientes flujos :

Flujo de Llegada.-
Lee resultado.

Flujo de Transacción.-
Selecciona.

Los módulos dependientes del proceso de transacción son:

Rechaza transacción.
Escribe transacción.
Crea encabezado.

Estos módulos están encargados de aceptar o rechazar la información dando

como resultado la transacción lista para enviar, o en su caso el rechazo de la transacción.

En la figura 3.7 se muestra la estructura de programa del control de envío de transacción, la cual presenta características de transformación, los flujos que la constituyen son :

| | |
|---------------------------|---------------------------------|
| Flujo de Llegada.- | Recepción de la señal de envío. |
| Flujo de Transformación.- | Formación de Grupo Funcional. |
| Flujo de Salida.- | Salida Transacción. |



Figura 3.7 Estructura de programa derivada para el control de envío de transacciones.

En la figura 3.8 se muestra el diagrama de la estructura de la información del Rechazo de transacciones, la cual presenta características de transformación. Dicho diagrama está formado principalmente por el módulo Control de rechazo de transacción, el cual presenta los siguientes flujos:

Flujo de Llegada.- Recepción de la señal de rechazo.

Flujo de Transformación.- Constituido por el módulo :

Control de verificación transacción. El cual presenta flujo de transacción, está formado por:

Flujo de Llegada.- Lee segmento no válido.
Flujo de Transacción.- Selección de operación.

Los módulos ligados a este flujo de transacción son :

Señal de envío de transacción
Búsqueda por fecha/hora.

Flujo de Salida.- Reporta resultados.

El cual presenta flujo de transacción, siendo los siguientes los flujos que deriva :

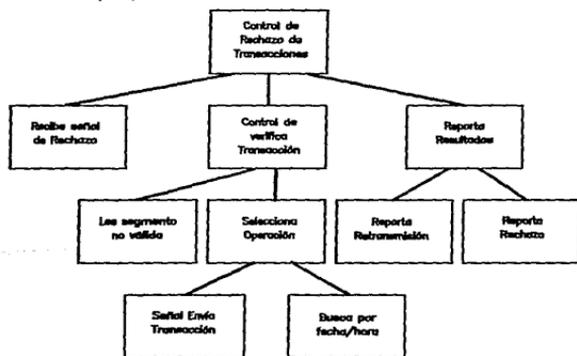


Figura 3.8. Estructura de programa derivada para el control de rechazo de transacciones.

Flujo de Llegada.-
Flujo de Transacción.-

Lee resultado.
Selección reporte.

Los módulos ligados son :

Reporta retransmisión.
Reporta rechazo.

Aplicando el análisis de transacción y transformación, así como las heurísticas de diseño, se pasó del diagrama de flujo de datos de la figura 3.4, al diagrama de estructura de programa de la figura 3.9.

En este diagrama se observan tanto flujos de transformación como flujos de transacción, y se detallan a continuación.

En el módulo de traductor de transacciones presenta una estructura de flujo de transformación, y para representarla adecuadamente, se generó un módulo de control llamado validación, con el fin de disminuir el abanico de salida y la profundidad.

El módulo de validación se compone de:

Flujo de Llegada.- Parser.

Flujo de Transformación.- Valida Clave de Acceso

Flujo de Salida.- Decisión de Validación

El módulo de Decisión de Validación fué generado debido a que en esa parte del diagrama de flujo de datos se presenta un flujo de transacción, en el cual se debe tomar la decisión de llamar al módulo de Analizar Información o llamar al de Reporte de Error.

Sistema Traductor de Intercambio Electrónico de Datos

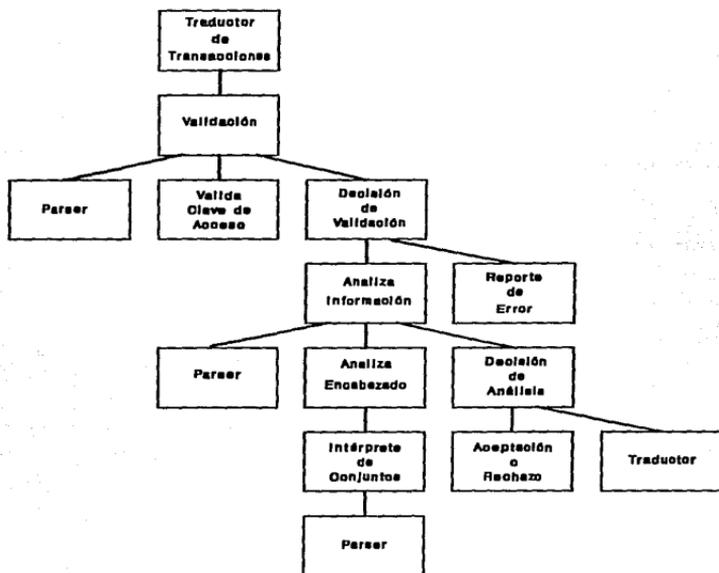


Figura 3.9 Estructura de programa derivada para la traducción de transacciones.

El módulo de Analiza Información presenta una estructura de flujo de Transformación y está compuesto de:

| | |
|---------------------------|---|
| Flujo de Llegada.- | Parser. |
| Flujo de Transformación.- | Analiza Encabezado, Intérprete de Conjuntos y Parser. |
| Flujo de Salida.- | Decisión de Análisis. |

El módulo de Decisión de Análisis fué generado para representar la estructura de transacción existente en esta parte del diagrama de flujo de datos, y define si se ejecuta el proceso de Aceptación o Rechazo, o el de Traductor.

El diagrama tiene una profundidad y anchura apropiada, así como abanicos de entrada y salida de uno, con excepción del parser, que es módulo de servicio, debido a que es utilizado por tres procesos.

3.3.3 CARTA DE ESTRUCTURA DE SOFTWARE

Carta de Estructura del Software

La carta de estructura del software es aquella en donde se muestra la información que se transfiere entre módulos en la estructura de programa, así como las iteraciones y las decisiones en las llamadas a los módulos subordinados.

También se pueden representar los procesos que se ejecutan en forma paralela y los módulos predefinidos, utilizando una estructura jerárquica definida por Constantine.

En la figura 3.10 se muestra la simbología utilizada en las figuras de carta de estructura de software.

Todos los diagramas de estructura de programas mostrados anteriormente, son presentados en formato de carta de estructura de software en las figuras 3.11, 3.12, 3.13 y 3.14. En esas figuras se presentan los flujos de datos entre módulos, así como aquellos que se ejecutan varias veces, y las decisiones que deben tomarse para procesar los módulos subordinados.

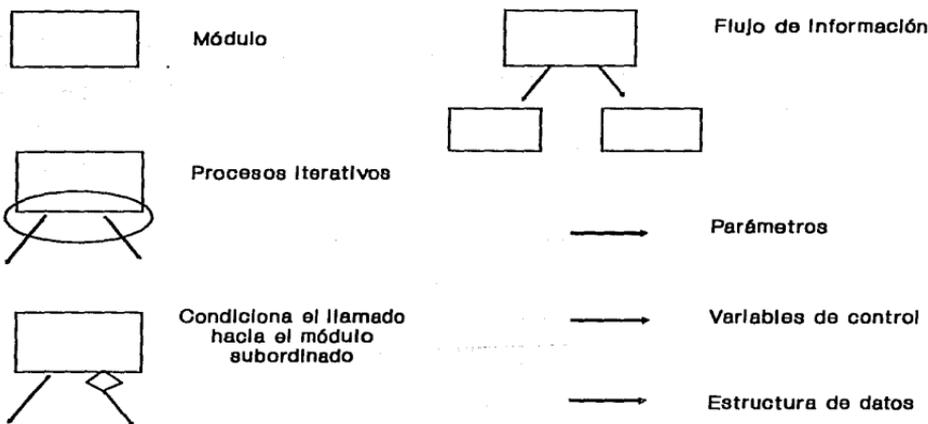


Figura 3.10. Simbología utilizada en la carta de estructura de software.

Sistema Traductor de Intercambio Electrónico de Datos

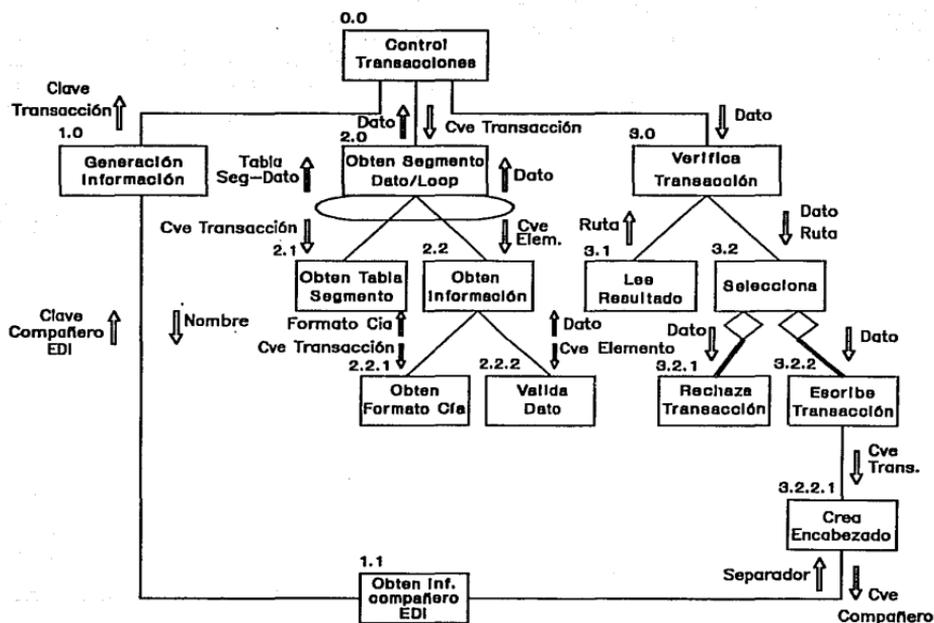


Figura 3.11. Carta de estructura de *software* de la generación de transacciones.

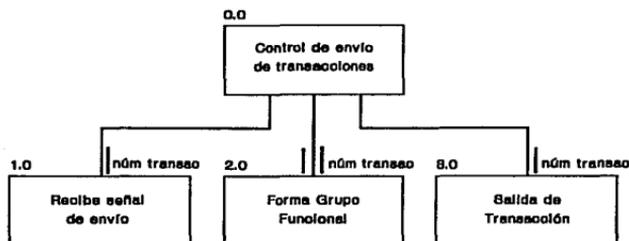


Figura 3.12. Carta de estructura de *software* del control de envío de transacciones.

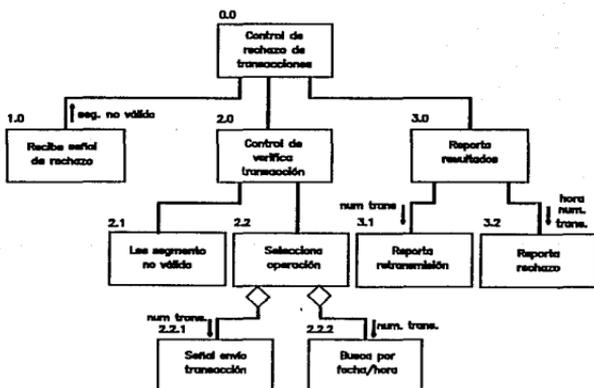


Figura 3.13. Carta de estructura de *software* del control de rechazo de transacciones.

Sistema Traductor de Intercambio Electrónico de Datos

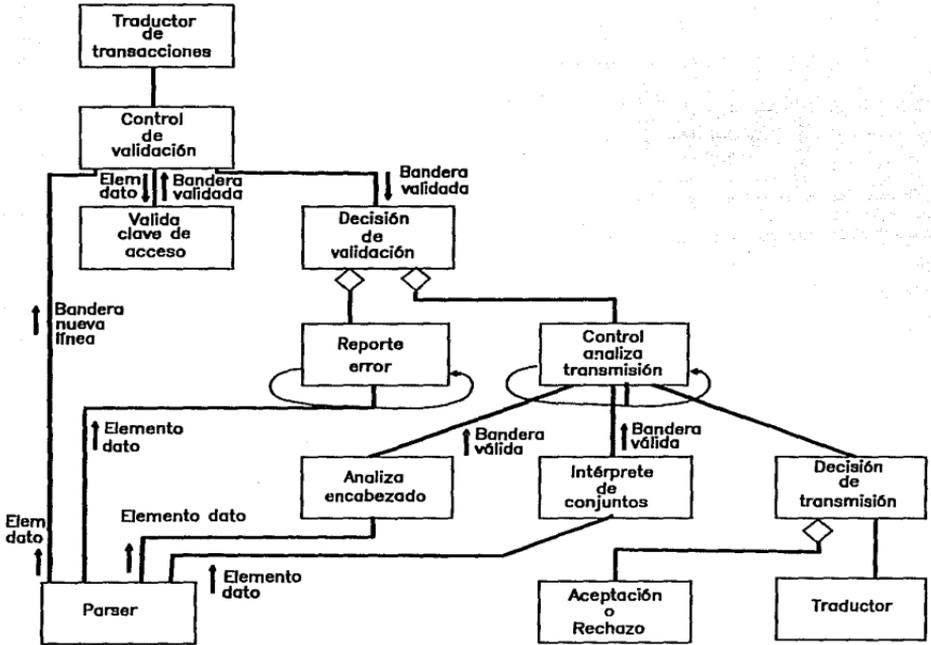


Figura 3.14. Carta de estructura de *software* de la traducción de transacciones.

3.4. DESCRIPCIÓN DE MÓDULOS

A continuación se presentan tres ejemplos de descripción de módulos, y son de los más importantes en el desarrollo de este trabajo.

MODULO DEL PARSER

3.4.1 Objetivo del módulo.

Extraer un elemento dato de una cadena de caracteres continua de información, almacenada en el archivo de mensajes EDI.

3.4.2 Información de entrada y salida.

| | |
|---|---------|
| Petición de elemento dato | Entrada |
| Caracteres de transacción del archivo de mensajes | Entrada |
| Caracter de separación | Entrada |
| Elemento dato | Salida |
| Bandera de CR/LF | Salida |

3.4.3 Procesos.

Leer de un archivo de mensajes los caracteres que conforman un elemento dato, almacenarlo en una estructura de datos y enviarlo a los módulos a los que atiende.

3.4.4 Especificación en un lenguaje de diseño.

INICIO

Elemento_Dato = ''

Bandera_CRLF = falso

Caracter = lee_caracter(archivo_mensajes)

```
Mientras (Caracter caracter_separador) Y
(NO_FIN_DE_LINEA) realiza
Inicio
Elemento_dato = Elemento_dato + caracter
Caracter = lee_caracter(archivo_mensajes)
fin
SI FIN_DE_LINEA ENTONCES
bandera_CRLF = verdadero
FIN
```

3.4.5 Archivos de entrada/salida

Archivo de mensajes EDI (TETRANS) Entrada

MODULO DE VALIDACION DE CLAVE DE ACCESO

3.4.1 Objetivo del módulo.

Validar la clave de acceso de la compañía que envía las transacciones EDI.

3.4.2 Información de entrada y salida.

| | |
|--|---------|
| Petición de validación | Entrada |
| Elemento dato de clave de acceso | Entrada |
| Elemento dato de código de compañero | Entrada |
| Clave de acceso catálogo compañero EDI | Entrada |
| Bandera de válido/inválido | Salida |

3.4.3 Procesos.

Llevar a cabo una comparación entre la clave de acceso del archivo de mensajes y la del archivo de compañeros EDI, y dependiendo del resultado, generar la bandera de válido/inválido.

Sistema Traductor de Intercambio Electrónico de Datos

3.4.4 Especificación en un lenguaje de diseño.

INICIO

DESDE I = 1 HASTA N REALIZA

 parser(elemento_dato,caracter_separador,bandera_C
 RLF)

 código_compa_EDI =
 parser(elemento_dato,caracter_separador,bandera_CRLF)

 clave_acceso_transmisión =
 parser(elemento_dato,caracter_separador,bandera_CRLF)

 clave_acceso_EDI = lee_clave(código_compa_EDI)

 SI clave_acceso_transmisión = clave_acceso_EDI
 ENTONCES

 bandera_válido = verdadero

 DE LO CONTRARIO

 bandera_válido = falso

FIN

3.4.5 Archivos de entrada/salida

Archivo de Compañeros EDI (TECEDI) Entrada

MODULO DE GENERACION DE GRUPO FUNCIONAL

3.4.1 Objetivo del módulo.

Generar el encabezado del grupo funcional de transacción.

3.4.2 Información de entrada y salida.

Fecha del sistema Entrada

Hora del sistema Entrada

3.4.3 Procesos.

Con la información de la transacción, obtener el grupo funcional al que corresponde y con los datos auxiliares armar el encabezado de la transacción.

3.4.4 Especificación en un lenguaje de diseño.

Colocar GS seguido del separador

Accesar la tabla 479 con el número de transacción y obtener el código funcional seguido del separador del compañero EDI.

Accesar la tabla 142 y obtener el número correspondiente a la aplicación de envío.

Accesar la tabla 124 y obtener el número correspondiente a la aplicación de recepción.

Obtener el número consecutivo de la tabla 28.

Colocar como código TDCC

Accesar la tabla 480 para obtener el código identificador de versión estándar.

3.4.5 Archivos de entrada/salida

TECEDI Entrada

TECTRTR Entrada

TRANSACCIONES Entrada/Salida

3.5. ESTRUCTURA DE ARCHIVOS Y DATOS GLOBALES

La estructura de los archivos se definió en la etapa de análisis y estas entidades no sufrieron modificaciones, por lo que para conocer su contenido es necesario consultar el capítulo de análisis.

3.6. PROVISIONES DE PRUEBA.

En la planeación de pruebas se describen sus objetivos, los criterios de aprobación, el plan de integración (estrategia, calendario, personal responsable) las herramientas particulares y las técnicas a utilizar, así como los casos reales y los resultados esperados. Las pruebas funcionales y de desempeño se desarrollan durante el análisis de requerimientos, son refinadas durante la fase de diseño. Las pruebas que examinan la estructura interna de un producto de programación y las que intentan destruir al sistema, llamadas también pruebas de tensión, se desarrollan durante el diseño detallado y durante la instrumentación.

La prueba de integración que se realizará en la tesis constará de los siguientes puntos:

Probar por separado cada uno de los módulos que constituyen al sistema.

Generar para cada uno de ellos los datos suficientes para probar todas las opciones presentadas en las especificaciones (a esta prueba se le llama prueba de la Caja Negra), asegurándose que por lo menos se pasa una vez por todos los caminos de cada módulo, se ejercitan todas las decisiones lógicas en sus caras verdaderas y falsas, se ejecutan las estructuras internas para asegurar su validez (prueba de la Caja Blanca).

esta página fué dejada en blanco intencionalmente

CAPITULO 4

desarrollo

4.1 CRITERIOS PARA SELECCIONAR EL LENGUAJE Y LAS HERRAMIENTAS DE DESARROLLO.

4.1.1 EL PROCESO DE TRADUCCION.

El paso de codificación traduce una representación del *software*, dada por un diseño detallado, a una realización en un lenguaje de programación. El proceso de traducción continúa cuando un compilador acepta el código fuente como entrada y produce como salida un código objeto, dependiente de la máquina; la salida del compilador es traducida a código de máquina.

4.1.2 CARACTERISTICAS DE LOS LENGUAJES DE PROGRAMACION.

Las características del lenguaje tiene un impacto directo sobre la calidad y la eficiencia de la traducción así como un impacto importante sobre el éxito de un proyecto y la calidad del mismo.

Las siguientes características aparecen como resultado del diseño de un lenguaje de programación:

La uniformidad. Indica el grado en que un lenguaje usa una notación consistente, aplica restricciones aparentemente arbitrarias o incluye excepciones a reglas sintácticas o semánticas.

La ambigüedad. Siempre es percibida por el programador. Un compilador siempre interpreta una sentencia de una misma forma, pero el lector humano puede interpretar la sentencia de formas diferentes.

Lo compacto. Es un indicativo de la cantidad de información orientada al código que se debe retener en la memoria humana. Los atributos que miden lo compacto de un lenguaje se muestran a continuación.

- grado en que soporta construcciones estructuradas
- tipos de palabras claves y abreviaturas que se pueden utilizar
- variedad de tipos de datos y características implícitas
- número de operadores aritméticos y lógicos
- número de funciones incorporadas

4.1.3 UN MODELO SINTACTICO/SEMANTICO

Cuando un programador aplica los métodos de la ingeniería de *software* que son independientes del lenguaje de programación aprovecha el conocimiento semántico. Por otro lado, el conocimiento sintáctico es dependiente del lenguaje, centrándose en las características de un lenguaje específico.

De estos tipos de conocimiento, el conocimiento semántico es el más difícil de adquirir y el más importante, intelectualmente, de aplicar. Todos los pasos de la ingeniería del *software* que preceden a la codificación hacen un fuerte uso del conocimiento semántico. El paso de codificación aplica el conocimiento sintáctico debido a que es "arbitrario e instructivo" y aprendido de forma rutinaria. Cuando se aprende un lenguaje nuevo de programación, se añade a la memoria nueva información sintáctica. Puede aparecer cierta confusión cuando la sintaxis de un nuevo lenguaje de programación es similar pero no equivalente a la sintaxis de otro lenguaje. Sin embargo, se debe hacer notar que un nuevo lenguaje de programación puede servir para motivar al ingeniero del *software* a aprender también su nueva información semántica.

4.1.4 UNA VISION DE INGENIERIA.

Una visión de ingeniería del *software*, sobre las características de los lenguajes de programación, se centra en las necesidades que puede tener un proyecto específico de desarrollo de *software*.

Sistema Traductor de Intercambio Electrónico de Datos

El conjunto general de características de Ingeniería para el código fuente son:

- 1) Facilidad de traducción del diseño al código;
- 2) Eficiencia del compilador;
- 3) Portabilidad del código fuente;
- 4) Disponibilidad de herramientas de desarrollo, y
- 5) Facilidad de mantenimiento.

El paso de codificación comienza tras haber definido, revisado y modificado en caso necesario el diseño detallado. En teoría, la generación de código fuente, a partir de las especificaciones del diseño detallado, debería ser algo directo. El grado de facilidad de la traducción del diseño al código proporciona una indicación de cómo se aproxima un lenguaje de programación a la representación del diseño.

Aunque los rápidos avances en la velocidad del procesador y la densidad de memoria han comenzado a disminuir la necesidad de "código super-eficiente", muchas aplicaciones todavía requieren programas rápidos y que utilicen poca memoria. Si el rendimiento del software es un requerimiento crítico, los lenguajes con compiladores optimizados pueden resultar más atractivos.

La portabilidad del código fuente es una característica de los lenguajes de programación que se puede interpretar de tres formas:

- 1.- El código fuente puede ser transportado de un procesador a otro y de un compilador a otro sin ninguna o muy pocas modificaciones.
- 2.- El código fuente permanece inalterado cuando cambia su entorno de funcionamiento. Por ejemplo, cuando se instala una nueva versión de un sistema operativo.

- 3.- El código fuente puede ser integrado en diferentes paquetes de software sin que prácticamente se requieran modificaciones debidas a las características propias del lenguaje de programación.

De las tres interpretaciones de portabilidad la primera es con mucho la más frecuente. La estandarización por la ISO (Organización Internacional de Estándares y/o el ANSI, Instituto Nacional Americano de Estándares) continúa siendo el principal esfuerzo para la mejora de la portabilidad de los lenguajes de programación. Desgraciadamente, la mayoría de los diseñadores sucumben ante la urgente necesidad de proporcionar "mejores" posibilidades, aunque no estándar, a un lenguaje estandarizado. Si la portabilidad es un requerimiento crítico, se debe restringir el código fuente al estándar ISO o ANSI, aunque existan otras posibilidades.

La disponibilidad de herramientas de desarrollo puede acortar el tiempo requerido para la generación del código fuente y puede mejorar la calidad del código. Muchos lenguajes de programación pueden ser adquiridos con un conjunto de herramientas que incluyen: compiladores con depuradores, ayudas de formato para el código fuente, facilidades de edición incorporadas, herramientas para control del código fuente, y otras.

La facilidad de mantenimiento del código fuente es críticamente importante para cualquier esfuerzo no trivial de desarrollo de software. El mantenimiento no se puede llevar a cabo hasta que no se entienda el software. Elementos anteriores de configuración del software (por ejemplo, documentación del diseño) proporcionan un fundamento para la facilidad de comprensión, ya que el código fuente final debe ser leído y modificado de acuerdo con los cambios en el diseño. Además, las propias características de documentación de un lenguaje tienen una fuerte influencia sobre el mantenimiento.

4.1.5 ELECCION DE UN LENGUAJE.

La elección de un lenguaje de programación para un proyecto específico debe tener en cuenta tanto las características de Ingeniería como las psicológicas. Sin embargo, el problema asociado con la elección puede aparecer si sólo se dispone de un lenguaje o si el cliente demanda uno en particular.

Entre los criterios que se aplican durante la evaluación de los lenguajes disponibles están:

- 1) Área de aplicación general;
- 2) Complejidad algorítmica y computacional;
- 3) Entorno en el que se ejecutará el *software*;
- 4) Consideraciones de rendimiento;
- 5) Complejidad de las estructuras de datos;
- 6) Conocimiento de la plantilla de desarrollo de *software*, y
- 7) Disponibilidad de un buen compilador o compilador cruzado.

El área de aplicación de un proyecto es el criterio que más se aplica durante el proceso de selección del lenguaje.

A menudo C es el lenguaje elegido para el desarrollo de *software* de sistemas, mientras que lenguajes como Ada, C y Modula-2 (junto con FORTRAN y ensamblador) se encuentran en aplicaciones de tiempo real. COBOL es el lenguaje para aplicaciones de negocios, aunque el uso creciente de lenguajes de cuarta generación puede algún día desplazarlo de su posición privilegiada. En el área científica/Ingeniería, FORTRAN permanece como el lenguaje predominante (aunque ALGOL, PL/1 y Pascal tienen un uso extendido). Las aplicaciones de Inteligencia Artificial usan lenguajes como el Lisp o PROLOG, aunque igualmente se usan lenguajes más convencionales.

4.1.6 LENGUAJES DE PROGRAMACIÓN E INGENIERÍA DE SOFTWARE.

Durante el paso de planificación del proyecto raramente se toman en consideración las características técnicas de un lenguaje de programación. Sin embargo, la planificación de las herramientas de soporte asociadas con la definición de recursos puede requerir que se especifique un compilador en particular (y su *software* asociado) o un entorno de programación. La estimación de costos y del plan de trabajo puede requerir que se ajuste la curva de aprendizaje debido a la Inexperiencia del personal con un determinado lenguaje.

Una vez que se han establecido los requerimientos del *software*, las características técnicas de los lenguajes de programación candidatos se hacen más importantes. Si se requieren estructuras de datos complejas, habrá que evaluar cuidadosamente los lenguajes que soporten estructuras de datos sofisticadas (por ejemplo, Pascal o Ada). Si lo importante es un alto rendimiento y posibilidades de tiempo real, se debe especificar un lenguaje diseñado para aplicaciones en tiempo real. (por ejemplo, Ada) o para eficiencia en memoria-velocidad (FORTH). Si se especifican muchos informes de salida y una fuerte manipulación de archivos se encontrarán lenguajes adecuados como COBOL o RPG.

El efecto de las características de un lenguaje de programación en los pasos que componen la prueba del *software* es difícil de asegurar. Los lenguajes que soportan directamente las construcciones estructurales tienden a reducir la complejidad ciclométrica de un programa, haciéndolo de alguna forma más fácil de probar. Los lenguajes que soportan la especificación de subprogramas y procedimientos externos hacen que la prueba de Integración sea mucho menos propensa a errores.

Sistema Traductor de Intercambio Electrónico de Datos

4.1.7 EL ESTILO DE CODIFICACION.

Una vez generado el código fuente, la función que realiza cada módulo debe resultar clara sin necesidad de referirse a ninguna fuente de especificación; es decir, el código debe ser comprensible. El estilo de codificación conlleva una filosofía de codificación que mezcle simplicidad con claridad.

Los elementos de estilo incluyen la documentación intensa, los métodos de declaración de datos, la aproximación a la construcción de sentencias y las técnicas de E/S.

4.1.7.1 DOCUMENTACION DEL CODIGO.

La documentación interna del código fuente comienza con la elección de los nombres de los identificadores, continúa con la localización y la composición de los comentarios y termina con la organización visual del programa.

La elección de nombres de identificadores significativos es crucial para la inteligibilidad. Los lenguajes que limitan la longitud de éstos implícitamente limitan la comprensión.

Obviamente hay que aplicar el sentido común al seleccionar los identificadores ya que los identificadores innecesariamente largos pueden ser una fuente potencial de error. Sin embargo, la elección de identificadores significativos mejora la comprensión.

La posibilidad de expresar los comentarios en lenguaje natural como parte del listado del código fuente es algo que aparece en todos los lenguajes de propósito general.

Esto se debe a que el *software* debe contener documentación interna. Los comentarios permiten al programador comunicarse con otros lectores del código fuente, y convertirse así en una clara guía de comprensión para la fase de mantenimiento.

Existen dos categorías de comentarios: los comentarios de prólogo y los comentarios descriptivos.

Los comentarios de prólogo deben aparecer al principio de cada módulo. El formato para estos es:

- 1.- Una sentencia que indique la función del módulo.
- 2.- Una descripción de la interfaz que incluya:
 - a) un ejemplo de "secuencia de llamada"
 - b) una descripción de todos los argumentos
 - c) una lista de todos los módulos subordinados
- 3.- Una discusión de los datos pertinentes.
- 4.- Una historia del desarrollo que incluya:
 - a) el diseñador del módulo
 - b) fecha de revisión y persona que la realizó
 - c) fechas de modificación

Los comentarios descriptivos se incluyen en el cuerpo del código fuente y se usa para describir las funciones del procesamiento. Estos comentarios deben:

- describir los bloques de código en lugar de comentar cada línea.
- usar líneas en blanco o tabulaciones de forma que sean fáciles de distinguir.
- ser correctos; un comentario incorrecto o que se pueda interpretar mal es peor que no ponerlo.

Por último, la forma en que el código aparezca en el listado es una importante contribución a la legibilidad. La indentación del código fuente realiza las construcciones lógicas y los bloques de código, tabulando desde el margen izquierdo de forma que se vean desplazados esos atributos.

4.1.7.2 DECLARACION DE DATOS.

El estilo en la declaración de datos se establece cuando se genera el código.

El orden de las declaraciones de datos se debe estandarizar, aún cuando el lenguaje de programación no tenga requerimientos específicos.

Cuando se declaran múltiples nombres de variables en una sola sentencia merece la pena ponerlos en orden alfabético.

4.1.7.3 CONSTRUCCION DE SENTENCIAS.

La construcción de sentencias individuales es parte del paso de codificación. La construcción de sentencias se basa en una regla general: cada sentencia debe ser simple y directa; el código no debe ser retorcido aunque se precise una mayor eficiencia.

Las sentencias de código fuente individuales se pueden simplificar al:

- evitar el uso de comparaciones condicionales complicadas,
- eliminar las comparaciones con condiciones negativas,
- evitar un gran anidamiento de bucles de condiciones,
- usar paréntesis para clarificar las expresiones lógicas aritméticas,
- usar espacios y/o símbolos claros para incrementar la legibilidad del contenido de la sentencia y,
- usar sólo características estándar ANSI

4.1.7.4 ENTRADA/SALIDA

El estilo de la entrada y la salida se establece durante el análisis de requerimientos de *software* y el diseño. Sin embargo, la forma en que se implementa la E/S puede ser una característica determinante de la aceptación del sistema por una comunidad de usuarios. El estilo de la entrada y salida variará con el grado de interacción humana. Para una E/S orientada a lotes serán deseables características tales como una organización lógica de la entrada, una comprobación de errores de entrada/salida significativa, una buena recuperación de errores de E/S y unos formatos de informes de salida racionales. Para la E/S interactiva lo principal será un esquema de entrada simple y dirigido, una extensa comprobación y recuperación de errores, una salida humanizada y una consistencia de formatos de E/S.

Dejando a un lado la naturaleza interactiva o no del *software*, se debe considerar una serie de principios de estilo de E/S durante el diseño y la codificación:

- 1.- Validar todos los datos de entrada.
- 2.- Comprobar las importantes combinaciones plausibles de elementos de entrada.
- 3.- Mantener el formato de entrada simple.
- 4.- Usar indicativos de fin-de-dato, en lugar de requerir al usuario que especifique el número de elementos.
- 5.- Etiquetar las peticiones interactivas de entrada, especificando las opciones posibles o valores límite.
- 6.- Mantener el formato de entrada uniforme cuando un lenguaje de programación tenga estrictos requerimientos de formato.
- 7.- Etiquetar todas las salidas y diseñar todos los informes.

Sistema Traductor de Intercambio Electrónico de Datos

El estilo de la E/S se ve afectado por otras muchas características tales como los dispositivos de E/S (p. ej., tipo de terminal, dispositivo de gráficos, ratón, etc...), la sofisticación del usuario y el entorno de comunicación. Wasserman da un conjunto extenso de principios para el "usuario de la Ingeniería del software y el diseño de sistemas interactivos". Estos principios, aplicables tanto al diseño de software como a la codificación, se resumen a continuación:

- 1.- Hacer invisibles al usuario los aspectos internos de la computadora
- 2.- Hacer que el programa sea a "prueba de balas", es decir, hacer que sea virtualmente imposible que el usuario haga que el programa termine de forma anormal
- 3.- Advertir al usuario cuando una petición pueda tener mayores consecuencias
- 4.- Proporcionar asistencia interactiva en el uso del programa
- 5.- Adecuar los requerimientos de entrada a las posibilidades del usuario
- 6.- Adecuar los mensajes de salida a la velocidad de los dispositivos de salida
- 7.- Distinguir entre distintas clases de usuarios
- 8.- Mantener un tiempo de respuesta consistente
- 9.- Minimizar el trabajo extra del usuario en los casos de error

Cada uno de estos principios debe llegar a ser un requerimiento implícito del software para todo un sistema interactivo. El software debe ser diseñado para que se ajuste a ellos y codificado de forma que implemente una buena interfaz de usuario bajo el punto de vista de la Ingeniería del software.

4.1.8 EFICIENCIA

En los sistemas catalogados como buenos, bajo el punto de vista de Ingeniería, existe una tendencia natural a usar los recursos críticos de forma eficiente. Los ciclos de procesador y las posiciones primarias de memoria a menudo se tratan como recursos críticos, y el paso de codificación se considera el último punto donde se le pueden arrancar segundos o bits al software. Aunque la eficiencia es un fin recomendable, se deben establecer tres máximas antes de pasar a una discusión más profunda. En primer lugar, la eficiencia es un requerimiento de rendimiento y, por tanto, se debe establecer durante el análisis de requerimientos del software. El software debe ser tan eficiente como se requiera, no tan eficiente como sea humanamente posible. En segundo lugar, la eficiencia se incrementa con un buen diseño. En tercer lugar, la eficiencia del código y su simplicidad van de la mano. En general, no hay que sacrificar la claridad, la legibilidad o la corrección en aras de unas mejoras en eficiencias que no sean esenciales.

4.1.8.1 EFICIENCIA EN CODIGO

La eficiencia del código fuente está directamente unida a la eficiencia de los algoritmos definidos durante el diseño detallado. Sin embargo, el estilo de codificación puede afectar a la velocidad de ejecución y a los requerimientos de memoria. El siguiente conjunto de directrices se debe seguir siempre en el proceso de traducción del diseño detallado al código:

- Simplificar las expresiones aritméticas y lógicas antes de convertirlas en código
- Evaluar cuidadosamente los bucles anidados para determinar si se pueden sacar fuera de ellos algunas sentencias o expresiones
- Cuando sea posible, evitar el uso de arreglos multidimensionales

- Cuando sea posible, evitar el uso de punteros y listas complejas
- Usar operaciones aritméticas "rápidas"
- No mezclar tipos de datos, incluso aunque el lenguaje lo permita
- Usar cuando sea posible aritmética entera y expresiones Booleanas

Muchos compiladores incluyen opciones de optimización que generan automáticamente código eficiente al colapsar expresiones repetitivas, evaluar los bucles, usar aritmética rápida y aplicar otros algoritmos relacionados con la eficiencia. Para las aplicaciones en las que la eficiencia sea vital, tales compiladores son una herramienta de codificación fundamental.

4.1.8.2 EFICIENCIA EN MEMORIA

Las restricciones de memoria en el mundo de las grandes máquinas ("grandes sistemas") son generalmente algo del pasado. La gestión de memoria virtual proporciona a las aplicaciones de software un espacio de direcciones lógicas enorme. La eficiencia en memoria para tales entornos no se puede relacionar con el uso de la menor memoria posible. Mas bien se deben tener en cuenta para la eficiencia en memoria las posibilidades de "paginación" del sistema operativo. En general, la localización o el mantenimiento del código en un dominio funcional, mediante construcciones estructuradas, es un método excelente para reducir la paginación y por tanto incrementar la eficiencia.

Las restricciones de memoria en el mundo de los microprocesadores son algo muy real, aunque las memorias de alta densidad y de baja coste están evolucionando rápidamente. Si los requerimientos del sistema demandan minimizar la memoria (p. ej., un producto de gran volumen y bajo coste), se deben evaluar cuidadosamente los compiladores de lenguajes de alto orden para ver sus posibilidades de comprensión de memoria.

A diferencia de otras características de los sistemas, que se encuentran enfrentadas unas con otras, las técnicas para conseguir una eficiencia en tiempo de ejecución pueden llevar a veces a una eficiencia en memoria. Por ejemplo, al limitar el uso de arreglos tri o tetradimensionales se obtienen algoritmos de acceso a los elementos sencillos que son más rápidos y más cortos. De nuevo, la clave para la eficiencia en memoria es "mantenerlo simple".

4.1.8.3 EFICIENCIA EN LA ENTRADA/SALIDA

Cuando se discute la eficiencia se han de considerar dos clases de E/S: E/S dirigida a la persona o E/S dirigida a otro dispositivo. La entrada suministrada por un usuario y la salida producida para un usuario son eficientes cuando la información se puede suministrar o se puede comprender con el mínimo esfuerzo intelectual.

La eficiencia de la E/S dirigida a otro hardware es un punto extremadamente complicado. Desde el punto de vista de la codificación (y del diseño detallado), sin embargo, se pueden establecer algunas directrices sencillas para mejorar la eficiencia en la E/S:

- Debe minimizarse el número de peticiones de E/S
- Toda la E/S debe ser buferada para reducir el embotellamiento en la comunicación
- Para las memorias secundarias (p.ej., discos), se debe seleccionar y usar el método de acceso más simple dentro de los aceptables
- La E/S a dispositivos de memoria secundaria debe hacerse por bloques
- La entrada salida a terminales e impresoras debe tener en cuenta las posibilidades del dispositivo que puedan afectar a la calidad o a la velocidad
- Recuerde que la "supereficiencia" en la E/S no merece la pena si no se puede comprender

Sistema Traductor de Intercambio Electrónico de Datos

4.1.9. CONSIDERACIONES SOBRE LA ELECCION DEL LENGUAJE DE PROGRAMACION Y EL EQUIPO UTILIZADO.

Uno de los objetivos planteados en los requerimientos del sistema es el que este sea portable, debido a que el concepto de Intercambio electrónico de datos se aplica en todas las ramas de la industria y se puede utilizar para transmitir datos entre empresas de muy diversa índole. Debido a lo anterior se evaluaron las características de diversos lenguajes de programación a utilizar y se analizaron las ventajas y desventajas de cada uno.

Los puntos de evaluación que se aplicaron durante la selección de los lenguajes de programación fueron:

- Que fueran estándares y que pudieran correr en varias plataformas de equipo de cómputo
- Que el equipo de desarrollo del software tuviera conocimientos adecuados del lenguaje
- Los recursos necesarios para ejecución
- El medio ambiente en el que se ejecutará el sistema
- Consideraciones de rendimiento
- Disponibilidad de un compilador con facilidades de depuración
- El área de aplicación del proyecto
- Manejo de estructuras de datos orientadas a cadenas de caracteres de longitud variable
- Manejo de archivos

La primera opción fue el lenguaje ensamblador, y fue descartada por la complejidad de programación, y por que no se requiere un tiempo de respuesta muy reducido. Además de que este lenguaje es inherente al procesador de cada equipo, con lo cual no existe la portabilidad de la aplicación.

La segunda, fue el lenguaje cobol, el cual tiene muchas ventajas para el manejo de archivos, puede manejar cadenas de caracteres de tamaño variable, y es sumamente portable entre varias plataformas de equipo. Las

desventajas son que la plantilla de desarrollo no lo conoce a fondo, que es un lenguaje muy antiguo, que no tiene un manejo amplio de estructuras de datos, y los programas manejan una cantidad considerable de código, con lo cual es más difícil de mantener. Por último, es recomendable que se lleve a cabo el cambio de cobol a lenguajes mas modernos para aprovechar los avances tecnológicos.

La siguiente opción fue el lenguaje pascal, el cual tiene las mismas ventajas de cobol con respecto al manejo de archivos y de cadenas de caracteres de longitud variable, pero además cuenta con un manejo muy extendido de estructuras de datos y es un lenguaje completamente estructurado, con un conjunto de instrucciones muy poderosas, lo cual hace que los programas sean reducidos en código y por lo tanto, más fáciles de mantener. Pascal se ejecuta en una amplia variedad de plataformas de hardware, y cuenta con compiladores que permiten una depuración rápida y eficiente de los programas. Por otro lado, la plantilla de desarrollo tiene un conocimiento profundo del lenguaje. Las desventajas de esta opción son que el manejo de archivos cambia en cada plataforma, y las funciones pueden cambiar de nombre en los diferentes compiladores, pero las funciones utilizadas en el presente trabajo se encuentran en todos los compiladores revisados.

La última opción evaluada fueron los lenguajes de cuarta generación, y la desventaja más marcada en éstos, es que no existe aún un estándar en la industria que permita una amplia portabilidad de las aplicaciones. Aún cuando ya existen en el mercado lenguajes capaces de operar en diversas plataformas de equipo, los requerimientos de recursos por parte de estos lenguajes son muy considerables. Además, para el presente trabajo se requiere la utilización de computadoras personales, y los requerimientos para ejecutar un lenguaje de cuarta generación en una microcomputadora están fuera del alcance del presente proyecto debido a los costos

asociados al equipo y al *software* que se utilizaría. Las ventajas de estos lenguajes son muy superiores a las ofrecidas por cualquiera de las tres opciones anteriores, con excepción de la portabilidad, ya que están orientados al desarrollo de aplicaciones de usuario final, cuentan con un manejo optimizado de archivos y estructuras de datos, son muy fáciles de mantener, sus instrucciones son muy poderosas, y manejan algo muy cercano al lenguaje natural.

Basados en los criterios descritos anteriormente, se tomó la decisión de utilizar el lenguaje pascal para el desarrollo del presente trabajo.

4.2 PRUEBAS

El desarrollo de sistemas de *software* envuelve una serie de actividades de producción en las que las posibilidades de que aparezca la fallibilidad humana son enormes. Los errores pueden empezar a darse desde el primer momento del proceso en el que los objetivos se pueden especificar de forma errónea o imperfecta, así como los errores que aparecen en los pasos posteriores de diseño y desarrollo. Debido a la imposibilidad humana de trabajar y comunicarse de forma perfecta, el desarrollo de *software* ha de ir acompañado de una actividad que garantice la calidad.

La prueba de *software* es un elemento crítico para la garantía de calidad del mismo y representa un último repaso de las especificaciones, del diseño y de la codificación.

4.2.1 OBJETIVOS DE LA PRUEBA

Los objetivos que se pretenden al realizar la prueba de *software* son:

- 1.- La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.

- 2.- Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
- 3.- Una prueba tiene éxito si descubre un error no detectado hasta entonces.

Los objetivos anteriores suponen un cambio dramático del punto de vista. Nos quitan la idea que normalmente se tiene de que una prueba tiene éxito si no descubre errores. El objetivo es diseñar pruebas que sistemáticamente saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y esfuerzo.



Figura 4.1. Diagrama del flujo de la información

4.2.2 FLUJO DE INFORMACIÓN DE LA PRUEBA

El flujo de información para la prueba sigue el esquema descrito en la figura 4.1.

Se proporcionan dos clases de entradas al proceso de prueba:

- 1.- Una configuración del *software* que incluye la especificación de requerimientos del *software*, la especificación del diseño y el código fuente; y
- 2.- Una configuración de prueba que incluye un plan y procedimiento de prueba, casos de prueba y resultados esperados.

Sistema Traductor de Intercambio Electrónico de Datos

Se lleva a cabo la prueba y se evalúan los resultados, o sea, se comparan los resultados de la prueba con los esperados. Cuando se descubren datos erróneos, esto implica que hay un error y comienza la depuración.

A medida que se van recopilando y evaluando los resultados de la prueba, comienza a vislumbrarse una medida cualitativa de la calidad y la fiabilidad del *software*. Si se encuentran con regularidad serios errores que requieren modificaciones en el diseño, la calidad de la fiabilidad del *software* quedan en entre dicho, siendo necesarias pruebas posteriores. Si por otro lado, el funcionamiento del *software* parece ser correcto y los errores que se encuentran son fácilmente corregibles, se puede sacar una de dos conclusiones:

- 1.- La calidad y la fiabilidad del *software* son aceptables.
- 2.- Las pruebas son inadecuadas para descubrir serios errores.

Finalmente, si la prueba no descubre errores, quedará la sospecha de que no se ha pensado cuidadosamente la configuración de prueba y de que los errores están escondidos en el *software*. Estos defectos serán eventualmente descubiertos por el usuario y corregidos por el profesional durante la fase de mantenimiento.

4.2.3 DISEÑO DE CASOS DE PRUEBA

Cualquier producto de ingeniería puede ser probado de una de dos formas:

- 1.- Conociendo la función específica para la que fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa;
- 2.- Conociendo el funcionamiento del producto se pueden desarrollar pruebas que aseguren que "todas las piezas encajan"; o sea, que la operación

interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada.

La primera aproximación de prueba se denomina "prueba de la caja negra" y la segunda "prueba de la caja blanca".

La prueba integral se realizará con una integración descendente en la cual se integran los módulos moviéndose hacia abajo por la jerarquía de control principal. Los módulos subordinados al módulo principal se van incorporando en la estructura por profundidad.

El proceso de integración se lleva a cabo en una serie de cinco pasos:

- Se usa el módulo de control principal como conductor de la prueba, disponiendo resguardos para todos los módulos directamente subordinados al módulo de control principal
- Se van sustituyendo los resguardos subordinados uno a uno por los módulos reales, la sustitución se realizará en profundidad (esto es iniciar y finalizar una rama)
- Se llevarán a cabo pruebas cada vez que se integra un nuevo módulo
- Tras terminar cada conjunto de pruebas, se reemplaza otro respaldo con el módulo real
- Se hace la prueba de regresión (o sea, todas o algunas pruebas anteriores) para asegurarse que no sean introducidos nuevos errores

Conjuntar todos los módulos y seguir la secuencia del flujo de datos para verificar:

- Que la información de interfaz entre módulos sea proporcionada/manejada correctamente,
- Que la escritura/lectura de los archivos se realice adecuadamente,
- Que la información captada/enviada sea la correcta

4.3 PRUEBAS REALIZADAS SOBRE EL SISTEMA.

Se llevó a cabo una prueba integral, de acuerdo a los criterios mencionados anteriormente. El desarrollo se generó por módulos, los cuales se iban insertando en el módulo de control principal, pero se trabajó en paralelo en los dos flujos de proceso principales que son la generación y la traducción de transacciones, respectivamente.

Sobre los módulos de captura de la información, se llevó a cabo la prueba de la caja negra debido a que el equipo de desarrollo no cuenta con los conocimientos necesarios sobre todas las variantes que pueden involucrar las transacciones presentadas. Sin embargo, para los módulos de la generación y la traducción de la información a formatos EDI, sí se realizaron las pruebas de la caja blanca.

Al final del desarrollo se procedió a probar el flujo del sistema, pasando por todos los caminos posibles de selección, e intentando hacer que los programas terminaran de manera anormal, lo cual se consiguió en algunos casos y se procedió a corregir los errores. Se revisaron los archivos y las salidas del sistema para verificar los resultados.

Después de lo anterior, se llevaron a cabo las pruebas de comunicación utilizando modems y transferencia de archivos, con el fin de simular un proceso de envío/recepción con un comercio EDI. Estas pruebas mostraron que en nuestro país este tipo de comunicación está limitada por la calidad de las líneas, debido a que la velocidad más alta que se puede alcanzar con los protocolos estándar es de 2400 bauds, pero puede funcionar apropiadamente cuando se utilizan modems que tengan manejo de corrección de errores, empaquetamiento de caracteres y un protocolo que maneje velocidades mayores dependiendo de la calidad de la línea.

esta página fué dejada en blanco intencionalmente

El aspecto más relevante que se desprende de este proyecto, es la necesidad de compartir información entre compañías de muy diversas índoles, evitando la recaptura de la misma, disminuyendo los errores, manejando información precisa de una manera oportuna, reduciendo los tiempos de proceso y de respuesta a los ciclos de producción o de prestación de servicios. Todas estas ventajas las proporciona el Intercambio Electrónico de Datos (EDI), un concepto en el cual realmente salen ganando todos los participantes.

El uso de estándares permite resolver y simplificar muchos de los problemas que se presentan actualmente en el manejo de la información. Hacen más comprensibles las cosas para todas aquellas personas que hagan uso de éstos, estableciendo de esta forma un "lenguaje" común entre ellos. También, facilitan la comunicación electrónica, con la que se logran diversos beneficios como optimización de tiempo, menor manejo de papel, etcétera.

Al tiempo que la tecnología avanza, las necesidades de comunicación utilizando medios magnéticos o electrónicos se van incrementando y la tendencia de estandarizar cobra aún mayor necesidad, por lo tanto no es conveniente mantenerse al margen. Es necesario integrarse a la evolución para poder influir en ella.

El Intercambio Electrónico de Datos tiene como objetivo el transmitir datos de manera electrónica, para que puedan ser procesados por una computadora utilizando formatos estándar.

Recordemos la definición de los estándares EDI que son:

Un conjunto de reglas, acordadas, aceptadas, y adquiridas voluntariamente, mediante las cuales los datos son estructurados en formatos de mensaje para intercambio de información operativa o de negocios.

Los estándares EDI han tenido gran aceptación en muchos países industrializados, los cuales pueden realizar aportaciones importantes para lograr implantar la utilización de los mismos en todo el mundo.

Al utilizar EDI se obtiene la capacidad de comunicarse con todas las empresas con las que se tienen tratos comerciales, utilizando el mismo sistema de cómputo y medios electrónicos, los cuales tienen la ventaja de ser muy rápidos, seguros, de un manejo cada vez más fácil, y al alcance de cualquier empresa. Lo anterior en comparación con lo que se viene haciendo en la actualidad, es un salto en concepto tan grande como lo fué la comunicación utilizando la telefonía, o la comunicación por medio del vídeo.

La utilización de los estándares EDI se está volviendo indispensable en cualquier compañía, principalmente en aquellas que tienen un número considerable de operaciones de negocios y en las que el intercambio de información con otras compañías es frecuente.

El uso de los estándares EDI presentan una serie de ventajas y desventajas, de las cuales se presentan algunas a continuación:

VENTAJAS

- * Se evita la recaptura de información debido a que ésta se registra una vez y es utilizada en los lugares donde se requiera.
- * Se disminuye el uso del papel para registrar la recepción y el envío de documentos, así como el manejo de copias de los mismos.
- * Se reduce el número de personal destinado al manejo de documentos.
- * Se reducen los costos del manejo de la información.
- * Se utilizan campos de longitud variable, lo que permite adaptarlos a las necesidades de cada una de las empresas.

Sistema Traductor de Intercambio Electrónico de Datos

- * La detección de datos erróneos se lleva a cabo fácilmente, debido a que antes de la transmisión y recepción de una transacción, la información es validada contra las características con que debe cumplir cada uno de sus campos, y es información de "primera mano".
- * Para realizar la corrección a alguna de las transacciones enviadas únicamente es necesario efectuar la modificación sobre el dato erróneo y volver a enviar la información.
- * Permite estandarizar la información de todos los documentos manejados en la compañía.
- * Se reducen los tiempos requeridos para los procesos productivos o de prestación de servicios.

DESVENTAJAS

- * Es necesario adquirir o desarrollar un sistema que utilice los estándares EDI, y que además sea capaz de manejar la información de la compañía.
- * Es necesaria una inversión inicial considerable en equipo de cómputo y en la adquisición o desarrollo del sistema.
- * Se requieren asociaciones que den mantenimiento e impulso al manejo de los estándares.

Un punto importante que debe ser tomado en cuenta por las empresas o las compañías desarrolladoras de software, al momento de querer participar en EDI, es que los estándares están adecuados a una realidad que no es la de México, y que la manera de hacer negocios es diferente en cada uno de los países donde se quiera implantar EDI.

Un ejemplo claro de lo anterior, es que la longitud de los datos definidos en los estándares de los E.U. (ANSI e ISO) no se adecuan a las condiciones establecidas en nuestro país, como es el caso de los campos numéricos reales cuyas longitudes (en algunos casos de 8 posiciones, de las cuales dos se ocupan para decimales y uno para el punto decimal) no satisfacen nuestros requerimientos de manejo de cifras (tan solo es posible representar decenas de millar en estos campos). Existen además tablas asociadas a campos

mandatorios cuya información no es significativa o no existe en México, y que sin embargo son imprescindibles para la adecuada formulación de las transacciones.

Además, un hecho que podría situarnos en gran desventaja es el tratado de libre comercio, al firmarse éste nuestras empresas estarán sujetas a concertar sus negocios a través de EDI, puesto que las compañías tanto de Canadá como de Estados Unidos así lo hacen. Sin duda alguna, debemos esperar como requisito que impondrán las empresas extranjeras para establecer negocios con las nuestras el trabajar con los formatos estándar para el intercambio electrónico de información. La manera de anticiparnos a ello es definiendo nuestras necesidades para que de esta forma los estándares EDI se ajusten a nuestra realidad. Una manera de lograrlo es la participación activa en los comités que definen los estándares, además de comenzar a tomar experiencia en la implantación y manejo de los estándares entre empresas mexicanas.

Sería ideal que México contara con sus propios estándares EDI, sin embargo se sabe que la definición y mantenimiento de éstos es una labor ardua y de gran peso para el grupo que la desarrolle.

Por lo anterior, es necesaria la creación de organismos en los cuales se discutan las necesidades de las empresas, la información que requieren compartir y los grados de seguridad que se deseen aplicar sobre la misma. Es muy importante que se comience a trabajar en México sobre los estándares de Intercambio Electrónico de Datos que se utilizarán en el país, tomando en cuenta la experiencia de los países industrializados, pero no copiando fielmente los formatos, por que las ventajas de EDI se verían reducidas considerablemente, ya que no se ajustarían a nuestras situaciones de negocios. Las universidades deben tener un papel importante en el impulso y desarrollo de estos

Conclusiones

estándares, pero es indispensable la participación de los órganos gubernamentales y de la industria, con el fin de generar estándares que tengan el consenso de los interesados, que estén regidos por una legislación moderna y eficiente, que cumplan con las características técnicas que se requieren para la transmisión de datos, y que en realidad ayuden a mejorar la productividad de las empresas, y por lo tanto del país.

El desarrollo de este trabajo nos permitió aplicar los conocimientos adquiridos durante nuestra estancia en la Facultad de Ingeniería de la UNAM, en la realización de un sistema de vanguardia.

El propósito inicial del trabajo de tesis fue el desarrollar un sistema prototipo capaz de manejar los estándares EDI, con el fin de cubrir una necesidad de Ferrocarriles Nacionales de México (FNM).

La idea surgió de un requerimiento hecho a ese organismo por parte de una asociación americana, para que manejará estándares EDI de manera obligatoria a partir de enero de 1992.

El desarrollo de la tesis consistió en analizar totalmente el funcionamiento de los estándares EDI, para generar un sistema que procesara ocho transacciones diferentes que son:

| Transacción | Usuario |
|------------------------------|--------------------------------|
| Renta de Carros | (obligatoria para FNM en 1992) |
| Detalle de Carga y Facturaje | (FNM) |
| Memorandum de Adeudo | (FNM) |
| Requerimiento de Cotización | (General) |
| Cotización | (General) |
| Orden de Compra | (General) |
| Mensaje Administrativo | (General) |

Transferencia de archivos (General)

utilizadas como prototipos para mostrar la funcionalidad de los mismos.

La experiencia derivada del trabajo de tesis fue el entender que es lo que se persigue con la utilización de formatos estándares, para la transferencia de información, y el trabajo que se requiere para dar soporte a los mismos, al desarrollar un traductor/generador de transacciones EDI. Durante el desarrollo del presente trabajo se utilizaron las técnicas de la Ingeniería de *software* para el desarrollo de un sistema, los principios de la programación estructurada, la teoría de compiladores, el lenguaje de programación Pascal, la teoría de estructuras de datos, etc.

Por otro lado, el sistema puede ser implantado en equipos de cómputo diversos a partir del análisis que se presentó en el capítulo 2. También puede ser fácilmente transportado a otros equipos debido a que fue desarrollado en lenguaje Pascal. Además, se le pueden agregar fácilmente más transacciones, debido a que sólo se requiere actualizar la tabla de estados del Parser, y definir la tabla de segmentos y datos asociados a las mismas.

Se pueden aplicar mejoras a la implantación como es el caso de una interfaz más amigable, ayudas en línea, un manejo de más transacciones, llevar a cabo multiproceso, y que maneje la transmisión/recepción de archivos.

Al finalizar la tesis se ha logrado el objetivo planteado originalmente, que fue el desarrollo de un sistema que maneje los estándares EDI para cubrir una necesidad de FNM.

esta página fué dejada en blanco intencionalmente

Glosario

AIAG

Automotive Industry Action Group. Grupo Automotriz responsable de EDI en su industria.

ANSI

American National Standards Institute, es una organización conformada por miembros de la comunidad industrial y de negocios. ANSI es responsable del establecimiento de los estándares cruzados de la industria en los Estados Unidos. Uno de los comités de ANSI, el ACREDITED STANDARDS COMMITTEE (ASC) es responsable del estándar EDI X.12, estándar ampliamente divulgado en los Estados Unidos. Este grupo es reconocido como coordinador y mediador de intereses para asuntos de manejo de información en Estados Unidos, y en algunos casos en Canadá. ANSI, también es el representante norteamericano para ISO (International Standards Organization).

Area de encabezado

El área de encabezado, contiene información preliminar que pertenece a un documento completo como la fecha, nombre de la compañía, dirección, etc.

CEFIC

Conseil European des Federations de L'Industria Chimique. El Consejo de asociaciones químicas manufactureras de Europa localizadas en Bruselas. Patrocina la iniciativa europea de EDI.

CIDX

Chemical Industry Data Exchange. Estándar de intercambio de Datos para la Industria Química basado en X.12. Las CIDX también son utilizados para hacer referencia a los lineamientos propios de la misma industria.

CCITT

Commite Consultative International Telegraphic and Telephonic.

Compañero de negocios

Cualquier compañía que acuerda realizar negocios con otra utilizando EDI.

Conjunto transaccional

Un documento de negocios completo, como una factura, una orden de compra, un aviso de embarque es sinónimo del documento o mensaje.

Diccionario de datos

La publicación que define todos los elementos de datos que existen para un estándar. El diccionario de datos EDI más utilizado es el diccionario JEDI.

EDI

Electronic Data Interchange. Es el término utilizado para denominar al intercambio computador a computador de documentos de negocio en formato estándar.

EDIA

Electronic data Interchange Association. Esta organización no lucrativa aporta su esfuerzo para la estandarización de mensajes y la estructura de comunicación basada en Transactions Sets.

EDIFACT

EDI For Administration, Commerce and Trade. Es la organización dependiente de las Naciones Unidas responsable del estándar EDI internacional.

EEC

European Economic Commission.

Sistema Traductor de Intercambio Electrónico de Datos

EFT

Electronic Funds Transfer

Elemento de Dato

La unidad más pequeña de información en el estándar, equivalente a un campo.

Encabezado

El segmento que indica el inicio de un archivo a ser transmitido. Los encabezados son estructuras de control.

Generación

La traducción de datos del formato del emisor al formato estándar.

Grupo funcional

Es el agrupamiento de conjunto transaccional relacionados de la misma clase. En un grupo funcional "PO" (Orden de compra), pueden incluirse órdenes de compra, acuses funcionales de órdenes de compra, etc.

Identificador de segmento

Es un código predefinido que identifica un segmento.

Intercambio electrónico de datos

Es el término utilizado para denominar al intercambio computador a computador de documentos de negocio en formato estándar.

ISO

International Standards Organization. Responsable del desarrollo de estándares de comunicaciones internacionales.

JEDI

Join Electronic Data Interchange. Grupo que busca la adopción de terminología acordada y una sintaxis universal para los mensajes entre Europa y Norte América. Este término también es utilizado para el diccionario de datos conjunto.

MHS

Message Handling Service.

Segmento de datos

Una línea de datos en un estándar. Está conformado por elementos de datos, y es análogo a un registro.

Separador de elementos

Es un carácter en particular que es usado para separar los elementos contenidos en un segmento.

TALC

Textile/Apparel Linkage Council.

TDCC

Transportation Data Coordination Committee.

Terminador de segmentos

Es un carácter especial que indica el fin de una línea de datos ANSI X.12.

UCS

Uniform Communication Standar. Estándar EDI usado históricamente en la industria de abarrotos.

UNECE

Standard Electronic Messages.

VAN(S)

Value Added Network.

VICS

Voluntary InterIndustry Communication Standards.

WINS

Warehouse Information Network Standard.

X12

El estándar ANSI para el intercambio de documentos de negocio en formato electrónico.

X25

Estándar Internacional para comunicación por paquetes de datos (en México el servicio Telepac es X.25).

X400

Estándar Internacional para comunicación (en desarrollo) para transmisión de mensajes.

esta página fué dejada en blanco intencionalmente

APENDICE A

Segmentos de Datos

EJEMPLO DE SEGMENTOS DE DATOS

A continuación se presentan algunos ejemplos de como se presentan en los manuales los segmentos de datos. Se incluyen únicamente los correspondientes a las transacciones de 410 (factura y detalles de carga) y 850 (orden de compra). Para consultar los demás, referirse al manual de segmentos y elementos de datos: Volume IV. Data Segments and Data Elements. The Electronic Data Interchange Association. Febrero 1989.

410 FACTURA Y DETALLES DE CARGA

ST ENCABEZADO DE TRANSACCION

| | | | | | | | |
|----|---|-------------|-------|---|------------|-------|----|
| ST | * | ST01 | 141 | * | ST02 | 320 | NL |
| | | TRANSACCION | | | NO CONTROL | | |
| | | M ID | 03/03 | | M AN | 04/04 | |

B3 INICIO DE SEGMENTO PARA FACTURA

| | | | | | | | | | | | | | |
|----|---|----------------|-------|---|------------|-------|---|------------------|-------|---|---------------|-------|----|
| B3 | * | B302 | 70 | * | B304 | 140 | * | B306 | 12 | * | B307 | 193 | NL |
| | | NUMERO FACTURA | | | FORMA PAGO | | | FECHA MOVIMIENTO | | | SALDO A PAGAR | | |
| | | M AN | 01/22 | | M ID | 02/02 | | M DT | 03/08 | | M N2 | 01/08 | |

N7 DETALLES DE EQUIPO

| | | | | | | | | | | | | | |
|----|---|-------------------|-------|---|-----------------|-------|----|------|-------|---|--------------|-------|----|
| N7 | * | N707 | 206 | * | N702 | 207 | * | N705 | 187 | * | N706 | 232 | NL |
| | | SELLOS ORIGINALES | | | NUMERO EQUIPO | | | TARA | | | LIMITE CARGA | | |
| | | C AN | 01/04 | | M AN | 01/10 | | C NO | 03/08 | | C NO | 02/08 | |
| | | N708 | 163 | * | N712 | 307 | NL | | | | | | |
| | | VOLUMEN | | | DUEÑO DE EQUIPO | | | | | | | | |
| | | C R | 01/08 | | C ID | 01/04 | | | | | | | |

G4 SEGMENTO DE IDENTIFICACION DE ESCALAS

| | | | | | | | | | | | | | |
|----|---|---------------------|-------|---|---------------|-------|---|---------|-------|----|--|--|--|
| GA | * | G401 | 19 | * | G402 | 156 | * | G403 | 450 | NL | | | |
| | | NOMBRE DE LA CIUDAD | | | CODIGO ESTADO | | | MOTIVOS | | | | | |
| | | M AN | 02/19 | | M ID | 02/02 | | O AN | 06/08 | | | | |

M7 NUMEROS DE SELLOS

| | | | | |
|----|---|------------------|-------|----|
| M7 | * | M701 | 225 | NL |
| | | NUMERO DE SELLOS | | |
| | | M AN | 02/15 | |

N8 REFERENCIA DEL NUMERO DE GUIA

| | | | | | | | |
|----|---|----------------|-------|---|------------------|-------|----|
| N8 | * | N801 | 180 | * | N802 | 183 | NL |
| | | NUMERO DE GUIA | | | FECHA DE LA GUIA | | |
| | | M NO | 01/08 | | M DT | 06/08 | |

F9 ESTACION ORIGEN

| | | | | | | | |
|----|---|-----------------|-------|---|---------------|-------|----|
| F9 | * | F902 | 101 | * | F903 | 156 | NL |
| | | ESTACION ORIGEN | | | CODIGO ESTADO | | |
| | | M AN | 02/18 | | M ID | 02/02 | |

Sistema Traductor de Intercambio Electrónico de Datos

D9 ESTACION DESTINO

D9 * D902 300 * D903 156
ESTACION DESTINO CODIGO DEL ESTADO NL
M AN 02/19 M ID 02/02

F1 NOMBRE DEL REMITENTE

F1 * F101 459 NL
REMITENTE
M AN 02/30

F2 DIRECCION DEL REMITENTE

F2 * F201 297 NL
DIR. REMITENTE
M AN 01/30

D1 NOMBRE DEL CONSIGNATARIO

D1 * D101 459 NL
CONSIGNATARIO
M AN 02/30

D2 DIRECCION DEL CONSIGNATARIO

D2 * D201 297 NL
DIR. CONSIGNATARIO
M AN 01/30

L5 DESCRIPCION MARCAS Y NUMEROS

L5 * L502 79 * L505 103 NL
DESCRIP CARGA CODIGO EMPACADO
C AN 01/30 O ID 05/05

L0 LINEA DE ARTICULO, CANTIDAD Y PESO

L0 * L001 213 * L004 81 * L010 459 NL
NO. LINEA CARGA PESO DESCRIPCION
O NO 01/03 O R 01/06 O AN 02/25

L1 TARIFA Y CARGOS

L1 * L114 74 NL
VALOR DECLARADO
O N2 02/10

L3 PESO TOTAL Y CARGOS

L3 * L301 81 * L305 58 * L307 117 NL
PESO TOTAL CARGOS ANTICIPO
O R 01/06 O N2 01/06 O N2 01/09

SE TERMINACION DE TRANSACCION

SE * SE01 90 * SE02 329 NL
No. SEGUIMIENTOS INCL. No. CONTROL TRANS.
M NO 01/06 M AN 04/06

850 ORDEN DE COMPRA

ST ENCABEZADO DE TRANSACCION

ST * ST01 143 * ST02 328
 TRANSACCION * NUM CONT TRANS
 M ID 03/03 M AN 04/06

BEG INICIO DE SEGMENTO PARA ORDEN DE COMPRA

BEG * BEG01 353 * BEG02 02 * BEG03 324 * BEG06 323
 COD TRANS COMP * TIPO ORD COMP * NUMERO DE ORDEN * FECHA DE LA ORDEN
 M ID 02/02 M ID 02/02 M AN 01/22 M DT 06/06

NTE INSTRUCCIONES ESPECIALES

NTE * NTE02 3
 MENSAJE
 M AN 01/80

PER CONTACTO ADMINISTRATIVO

PER * PER01 306 * PER02 03
 COD PUESTO CONT * NOMBRE
 M ID 02/02 O AN 01/35

N1 NOMBRE

N1 * N101 06 * N102 03
 COD ENT FORM ORD * NOMBRE
 M ID 02/02 O AN 01/35

N3 DIRECCION

N3 * N301 106
 DIR ENT FORM ORD
 M AN 01/35

PKG REQUERIMIENTOS DE ARTICULOS

PKG * PKG06 352
 DESCRIP ARTIC
 C AN 01/80

PO1 DATOS DE LOS ARTICULOS

PO * PO102 330 * PO103 355
 CANTIDAD ORDENAR * UNIDAD DE MEDIDA
 O AN 01/06 M ID 02/02

CTP INFORMACION DE PRECIOS

CTP * CTP03 067
 PRECIO UNITARIO
 C R 01/14

AMT IMPORTE MONETARIO

AMT * AMT02 782 * NL
 IMPORTE MONETARIO
 M R 01/15

SE TERMINACION DE TRANSACCION

SE * SE01 06 * SE02 328
 NUM SEG INCLUIDOS * NUM DE CON TRANS
 M NO 01/06 M AN 04/06

esta página fué dejada en blanco intencionalmente

APENDICE B

Procedimientos

EJEMPLOS DE PROCEDIMIENTOS

A continuación se presentan algunos ejemplos de las rutinas mas importantes del sistema desarrollado. Las primeras están en lenguaje PASCAL y las siguientes en el lenguaje utilizado por DBASE.

```
(*****
****
**** MODULO      :  TRAD_CIA.PAS          NUMERO DE MODULO : 4 ****
****
**** FUNCION     :  Traduce las transacciones EDI a formatos ****
****              que maneja la compania.          ****
****
****              Analiza el grupo funcional consultando - ****
****              la base de datos EDI, para obtener tanto ****
****              los datos del companero como el tipo de ****
****              las transacciones que envuelve el grupo ****
****              funcional. Con estos datos se procede a ****
****              la traduccion de la transaccion generan- ****
****              do la informacion en los formatos de la ****
****              compania. Dependiendo de la validez de - ****
****              la transaccion se solicita la transmi- ****
****              sion del acuse de recibo o del rechazo - ****
****              de transaccion.                    ****
****
**** SUBROUTINAS ****
**** REFERENCIADAS : ANALIZA_ENCABEZADO          ****
****                 REPORTE_SEGURIDAD          ****
****                 ACEPTACION_RECHAZO       ****
****                 PARSER                      ****
****                 ANALIZA_ENCABEZADOS        ****
****                 INTERPRETE                 ****
****                 TRADUCTOR                  ****
****                 REPORTE_ESTADISTICAS       ****
****
**** AUTOR(ES)   :  Jesus M. Pastran Rodriguez ****
****              Susana Yanez Rayon          ****
****
*****)
```

Unit Rut_Cia;

Interface

Uses

Crt, Dos;

Sistema Traductor de Intercambio Electrónico de Datos

```
Procedure Analiza_Encabezado(Var Arc_Transac:Text);
```

```
Implementation
```

```
Function Parser(Transaccion:String;Var I:Integer):String;
```

```
{** FUNCION : Identifica cada elemento dato de acuerdo al caracte- **  
**          ter separador, en este caso dicho caracter es el caracte- **  
**          asterisco. **}
```

```
Var
```

```
Aux_Elemento : String; {Auxiliar para formar el elemento dato}
```

```
Begin
```

```
Aux_Elemento := ' '; Aux_Elemento:='';  
While (Transaccion[I] <> '*') and (I <= Ord(Transaccion[0])) do  
begin  
Aux_Elemento := Concat(Aux_Elemento,Transaccion[I]);  
I := I+1;  
end;  
If Transaccion[I] = '*' then  
I := I+1;  
Parser := Aux_Elemento;
```

```
End;
```

```
Procedure Interprete(Var Arc_Transac:Text; Num_Transac:Integer);
```

```
{** FUNCION : Verifica que cada elemento dato tenga las condiciones **  
**          y el orden establecidos en la base de datos EDI, ge- **  
**          nera la peticion de rechazo de transaccion en caso de **  
**          error. **}
```

```
Var
```

```
I, {Contador }  
Pos_Elemento, {Posicion del elemento dentro de la transaccion}  
Long_Elemento, {Longitud del elemento }  
Num_Segmentos, {Cuenta los segmentos para verificar el dato - }  
{que trae la transaccion. }  
Aux_Segmentos, {Auxiliar que guarda en una variable Integer - }  
{el elemento dato de numero de segmentos. }  
Codigo : Integer; {Guarda el codigo de conversion de string-integer}  
Segmento,  
Elemento_Dato,  
Nomb_Arch, {Variable para seleccionar en que archivo se }  
{debe escribir de acuerdo a la transaccion }  
Reg_Compania : String; {Registro en el formato de la compania }
```

Procedimientos

```
Transaccion      : Array [1..54] of Integer;  
                  {Vector de características del registro de la }  
                  {compañía de acuerdo a la transacción.      }  
Arch_Compania    : Text;  
Bandera          : Boolean; {Bandera booleana para determinar si se gene- }  
                  {ra transacción de aceptación o rechazo.      }  
                  }
```

Procedure Traductor;

```
{** FUNCION : Convierte una transacción EDI a formato interno de **  
**           la base de datos de la compañía, auxiliándose por **  
**           el arreglo de datos Transaccion, el cual contiene **  
**           las longitudes de los campos de la base de datos **  
**           de la compañía.                                     **}
```

```
Var  
Blancos : String;      {Auxiliar para completar el tamaño del campo}  
I       : Integer;    {Contador }
```

Begin

```
Blancos := '';  
If Long_Elemento < Transaccion[Pos_Elemento] then  
  For I:=1 to (Transaccion[Pos_Elemento] - Long_Elemento) do  
    Blancos := Blancos + ' '  
  Reg_Compania := Reg_Compania + Elemento_Dato + Blancos;  
End;
```

Procedure Carga_Transacciones;

```
{** FUNCION : Mapeo de longitud entre los elementos dato de las **  
**           transacciones y los campos de los formatos que - **  
**           utiliza la compañía.                               **}
```

Begin

```
Transaccion[ 1]:= 2;  Transaccion[ 2]:= 3;  Transaccion[ 3]:= 9;  
If Num_Transac = 410 then  
  begin  
    Transaccion[ 4]:= 2;  Transaccion[ 5]:= 6;  Transaccion[ 6]:= 2;  
    Transaccion[ 7]:= 6;  Transaccion[ 8]:= 9;  Transaccion[ 9]:= 2;  
    Transaccion[10]:= 4;  Transaccion[11]:=10;  Transaccion[12]:= 8;  
    Transaccion[13]:= 6;  Transaccion[14]:= 8;  Transaccion[15]:= 4;  
    Transaccion[16]:= 2;  Transaccion[17]:=19;  Transaccion[18]:= 2;  
    Transaccion[19]:=19;  Transaccion[20]:= 2;  Transaccion[21]:=15;  
    Transaccion[22]:= 2;  Transaccion[23]:= 6;  Transaccion[24]:= 6;  
    Transaccion[25]:= 2;  Transaccion[26]:=19;  Transaccion[27]:= 2;
```

Sistema Traductor de Intercambio Electrónico de Datos

```
Transaccion[28]:= 2; Transaccion[29]:=19; Transaccion[30]:= 2;
Transaccion[31]:= 2; Transaccion[32]:=30; Transaccion[33]:= 2;
Transaccion[34]:=30; Transaccion[35]:= 2; Transaccion[36]:=30;
Transaccion[37]:= 2; Transaccion[38]:=30; Transaccion[39]:= 2;
Transaccion[40]:=50; Transaccion[41]:=05; Transaccion[42]:= 2;
Transaccion[43]:= 3; Transaccion[44]:= 8; Transaccion[45]:=25;
Transaccion[46]:= 2; Transaccion[47]:=10; Transaccion[48]:= 2;
Transaccion[49]:= 8; Transaccion[50]:= 9; Transaccion[51]:= 9;
Transaccion[52]:= 2; Transaccion[53]:= 6; Transaccion[54]:= 9;
Nomb_Arch := 'TECIA410.DAT';
end
Else
begin
Transaccion[52]:= 2; Transaccion[53]:= 6; Transaccion[54]:= 9;
Nomb_Arch := 'TECIA850.DAT';
end;
End;
```

```
Begin
Carga_Transacciones; Reg_Compania := ' ';
Assign(Arch_Compania,Nomb_Arch);
Rewrite(Arch_Compania);
Pos_Elemento := 1; Num_Segmentos := 0;
While not EOF(Arc_Transac) do
begin
I:=1;
Readln(Arc_Transac,Segmento);
While I <= Ord(Segmento[0]) do
begin
Elemento_Dato := Parser(Segmento,I); {Obtiene el elemento dato}
Long_Elemento := Length(Elemento_Dato); {y su longitud }
{Si es un dato de la compania se obtiene }
{de lo contrario solo se checa su longitud }
If (Long_Elemento <= Transaccion[Pos_Elemento]) and
(Pos_Elemento in [5..8,10..15,17..19,21,23,24,26,27,29,30,32,
34,36,38,43..45,47,49..51]) then
Traductor
Else
{Si encuentra error solicita generacion de }
{transaccion de rechazo (Bandera=False) }
If Long_Elemento > Transaccion[Pos_Elemento] then
begin
Bandera := False;
Aceptacion_Rechazo(Bandera);
end
Else
```

Procedimientos

```
                {Verifica que el dato de numero de segmentos}
                {incluidos en la transaccion sea correcto, }
                {si existe error se rechaza y se solicita -}
                {transaccion de rechazo. }
If Pos_Elemento = 53 then
begin
  Val(Elemento_Dato,Aux_Segmentos,Codigo);
  If Aux_Segmentos <> Num_Segmentos then
  begin
    Bandera := False;
    Aceptacion_Rechazo(Bandera);
  end
  Else
  begin
    {Si todos los elementos datos fueron correc-}
    {tos se pide transaccion de acuse de recibo }
    {(Aceptacion, Bandera=True) }
    Bandera := True;
    Aceptacion_Rechazo(Bandera);
  end;
end;
  Pos_Elemento := Pos_Elemento + 1;
end;
  Num_Segmentos := Num_Segmentos + 1;
  Writeln(Arch_Compania,Reg_Compania);
end;
Close(Arch_Compania);
End;
```

Procedure Analiza_Encabezado(Var Arc_Transac:Text);

```
(** FUNCION : Analiza los dos diferentes encabezados de control, **
** por lo que requiere llamar iterativamente al modulo **
** Parser para que le proporcione elementos dato, los **
** cuales debe verificar contra la base de datos EDI. **
** En caso de error debera requerir reporte de seguri- **
** dad, de lo contrario se procede a la traduccion de **
** las transacciones **)
```

```
Type
  Caracteristicas = Record
    {Caracteristicas de los elemntos dato del}
    {encabezado. }
    Tipo_Elemento, {1-Alfanumerico, 2-Numerico, 3-Fecha/Hora}
    Longitud_Valida : Integer;
  end;
```

Var

Sistema Traductor de Intercambio Electrónico de Datos

```
Encabezado      : String[255];  {Variable que guarda los encabezados      }
Elemento_Dato   : String;       {Elementos Datos que regresa el Parser      }
I,              : Integer;       {Posicion dentro del encabezado            }
Long_Elemento,  : Integer;       {Auxiliar para saber la longitud del elemento}
Pos_Elemento,  : Integer;       {Posicion del elemento dato en el segmento }
Num_Transac    : Integer;       {Auxiliar para verificar el numero de tran- }
                                      {saccion.                                   }

Seg_Encabezado : Array [1..8] of Caracteristicas;
                                      {Vector que contiene las caracteristicas del }
                                      {elemento dato                               }
Bandera        : Boolean;       {Indica segmento valido o incorrecto       }
```

Procedure Carga_Encabezado;

```
{** FUNCION : Carga las caracteristicas de los elementos que forman **
**          : el segmento del encabezado BG                          **}
```

Begin

```
Seg_Encabezado[1].Tipo_Elemento := 1;
Seg_Encabezado[1].longitud_Valida := 2;
Seg_Encabezado[2].Tipo_Elemento := 1;
Seg_Encabezado[2].longitud_Valida := 10;
Seg_Encabezado[3].Tipo_Elemento := 1;
Seg_Encabezado[3].longitud_Valida := 10;
Seg_Encabezado[4].Tipo_Elemento := 1;
Seg_Encabezado[4].longitud_Valida := 12;
Seg_Encabezado[5].Tipo_Elemento := 1;
Seg_Encabezado[5].longitud_Valida := 12;
Seg_Encabezado[6].Tipo_Elemento := 3;
Seg_Encabezado[6].longitud_Valida := 6;
Seg_Encabezado[7].Tipo_Elemento := 3;
Seg_Encabezado[7].longitud_Valida := 4;
Seg_Encabezado[8].Tipo_Elemento := 2;
Seg_Encabezado[8].longitud_Valida := 5;
```

End;

Procedure Verifica_Encabezado;

```
{** FUNCION : Verifica que los elementos dato sean correctos, si lo **
**          : son (Bandera=True) pide generacion de aceptacion, de **
**          : lo contrario pide generacion de rechazo.                **}
```

Begin

```
If Long_Elemento <= Seg_Encabezado[Pos_Elemento].Longitud_valida then
  Bandera := True
```

```

Else
  Bandera := False;
End;

```

```

*****
* Titulo      : CONOCIMI.PRG
* Proposito   : Actualizar informacion de la transaccion de Detalles de
*              Carga y Factura.
* Ejemplo de llamada : DO CONOCIMI
*
* Autor (es)  : Ricardo G. Gonzalez G.
*              : Leticia Reyes Torres
*****

```

```

USE CONOCIMI
SET INDEX TO CONOCIMI
NUMSOLI = SPACE(6)
ANO      = "AA"
MES      = "MM"
DIA      = "DD"
APPEND BLANK
REGISTRO=RECNO()
DO WHILE .T.
  @ 0,0
  ?? CHR(255)+CHR(255)+"CONOCIMI/"
  ?? CHR(255)+CHR(255)+"F1/"
  @ 6,20 GET NUMSOLI  PICTURE "@61"
  @ 6,62 GET ANO     PICTURE "99"
  @ 6,65 GET MES     PICTURE "99"
  @ 6,68 GET DIA     PICTURE "99"
  @ 8,20 GET BNUMGUIA PICTURE "999999"
  @ 8,62 GET ANO     PICTURE "99"
  @ 8,65 GET MES     PICTURE "99"
  @ 8,68 GET DIA     PICTURE "99"
  @ 14,18 GET BESCALA PICTURE "@191"
  @ 17,18 GET BCODESCA PICTURE "99"
  @ 20,18 GET BMOTESCA PICTURE "@191"
  @ 12,58 GET BPROPIET PICTURE "@41"
  @ 14,58 GET BINICIAL PICTURE "@41"
  @ 14,63 GET BNUMEROS PICTURE "@101"
  @ 16,58 GET BCAPACID PICTURE "99999.99"
  @ 18,58 GET BTARA    PICTURE "99999.99"
  @ 20,58 GET BLIMCARG PICTURE "999999"
  @ 22,58 GET BSELLOS  PICTURE "@151"
CLEAR GETS
Set Color to N/W
@ 3,37 Say " ALTAS "

```

Sistema Traductor de Intercambio Electrónico de Datos

```
DO WHILE .T.
  GOTO REGISTRO
  @ 6,20 GET NUMSOLI  PICTURE "@6!"
  READ
  F1=READKEY()
  IF F1=36 .OR. F1=37
    DELETE
    PACK
    EXIT
  ENDIF
  SET COLOR TO
  SEEK NUMSOLI
  IF EOF()
    EXIT
  ELSE
    IF REGISTRO<>RECNO()
      SET COLOR TO W/R,,
      @ 24,30 SAY " LA CLAVE YA EXISTE"
      ? REPLICATE(CHR(7),3)
      SET COLOR TO
      @ 24,30 SAY "
    ELSE
      EXIT
    ENDIF
  ENDIF
ENDDO
IF F1=36 .OR. F1=37
  EXIT
ENDIF
GOTO REGISTRO
REPLACE BNUMSOLI WITH NUMSOLI
@ 6,62 GET BANOSOLI  PICTURE "99"
@ 6,65 GET BMESSOLI  PICTURE "99"
@ 6,68 GET BDIASOLI  PICTURE "99"
@ 8,20 GET BNUMGUIA  PICTURE "999999"
@ 8,62 GET BANOGUIA  PICTURE "99"
@ 8,65 GET BMESGUIA  PICTURE "99"
@ 8,68 GET BDIAGUIA  PICTURE "99"
@ 14,18 GET BESCALA  PICTURE "@19!"
@ 17,18 GET BCODESCA PICTURE "99"
@ 20,18 GET BMOTESCA PICTURE "@19!"
@ 12,58 GET BPROPIET  PICTURE "@4!"
@ 14,58 GET BINICIAL  PICTURE "@4!"
@ 14,63 GET BNUMEROS  PICTURE "@10!"
@ 16,58 GET BCAFACID  PICTURE "999999.99"
@ 18,58 GET BTARA     PICTURE "999999.99"
@ 20,58 GET BLIMCARG  PICTURE "999999"
```

Procedimientos

```
@ 22,58 GET BSELLOS PICTURE "@15!"
READ
STORE "S" TO RESPUESTA
SET COLOR TO N/W,,
@ 24,26 SAY "ES CORRECTA LA INFORMACION ? " GET RESPUESTA
READ
IF RESPUESTA$'Sa'
  @ 0,0
  ?? CHR(255)+CHR(255)+"CONOCI_1/"
  ?? CHR(255)+CHR(255)+"F1/"
  @ 8,18 GET BESTORIG PICTURE "@19!"
  @ 10,18 GET BCEESTOR PICTURE "99"
  @ 15, 7 GET BREMITEN PICTURE "@30!"
  @ 18, 7 GET BDIREMIT PICTURE "@30!"
  @ 8,58 GET BESTDEST PICTURE "@19!"
  @ 10,58 GET BCEESTDE PICTURE "99"
  @ 15,47 GET BCONSIGN PICTURE "@30!"
  @ 18,47 GET BDICONSI PICTURE "@30!"
  CLEAR GETS
  Set Color to N/W
  @ 3,37 Say " ALTAS "
  @ 8,18 GET BESTORIG PICTURE "@19!"
  @ 10,18 GET BCEESTOR PICTURE "99"
  @ 15, 7 GET BREMITEN PICTURE "@30!"
  @ 18, 7 GET BDIREMIT PICTURE "@30!"
  @ 8,58 GET BESTDEST PICTURE "@19!"
  @ 10,58 GET BCEESTDE PICTURE "99"
  @ 15,47 GET BCONSIGN PICTURE "@30!"
  @ 18,47 GET BDICONSI PICTURE "@30!"
  READ
  STORE "S" TO RESPUESTA
  SET COLOR TO N/W,,
  @ 24,26 SAY "ES CORRECTA LA INFORMACION ? " GET RESPUESTA
  READ
  IF RESPUESTA$'Sa'
    @ 0,0
    ?? CHR(255)+CHR(255)+"CONOCI_2/"
    ?? CHR(255)+CHR(255)+"F1/"
    SELECT 2
    USE CONOLOOP
    BANDERA = .T.
    RENG = 8
    DO WHILE BANDERA
      APPEND BLANK
      REPLACE BSOLICIT WITH NUMSOLI
      @ RENG, 4 GET BCVECARG PICTURE "999"
      @ RENG,16 GET BCNCARG PICTURE "@25!"
```

Sistema Traductor de Intercambio Electrónico de Datos

```
@ RENG,49 GET BPESCARG PICTURE "99999.99"
@ RENG,65 GET BCUOCARG PICTURE "9,999,999.99"
READ
F1=READKEY()
IF (F1=36 .OR. F1=37) .AND. RENG < 18
  DELETE
  PACK
  BANDERA = .F.
ELSE
  RENG = RENG + 1
ENDIF
ENDDO
USE
SELECT 1
@ 17,20 GET BPESODOC PICTURE "99999.99"
@ 19,20 GET BCONPAGA PICTURE "99"
@ 17,61 GET BIMPORIG PICTURE "999,999,999"
@ 19,61 GET BANTPAGA PICTURE "999,999,999"
@ 21,61 GET BSALDO PICTURE "999,999,999"
READ
STORE "S" TO RESPUESTA
SET COLOR TO N/W,,
@ 24,26 SAY "ES CORRECTA LA INFORMACION ? " GET RESPUESTA
READ
IF RESPUESTA$'S$'
  APPEND BLANK
  REGISTRO=RECNO()
  NUMSOLI = SPACE (6)
  ANO = "AA"
  MES = "MM"
  DIA = "DD"
ENDIF
ENDIF
ENDDO.
```

Bibliografía

*EDI FORUM.
THE JOURNAL OF ELECTRONIC DATA INTERCHANGE
EDI PUBLICATION, 1989*

*CONCEPT, VOLUME I
ELECTRONIC DATA INTERCHANGE ASSOCIATION
FEBRUARY 1989*

*RAIL TRANSPORTATION INDUSTRY APPLICATION, VOLUME III
ELECTRONIC DATA INTERCHANGE ASSOCIATION
AUGUST 1988*

*BUSSINES, VOLUMEN III
ELECTRONIC DATA INTERCHANGE ASSOCIATION
FEBRUARY 1989*

*DATA SEGMENT AND DATA ELEMENT, VOLUME IV
ELECTRONIC DATA INTERCHANGE ASSOCIATION
FEBRUARY 1989*

*STRUCTURED SYSTEMS ANALYSIS: TOOLS AND TECHNIQUES
CHRIS GANE AND TRISH SARSON
MCDONELL DOUGLAS PROFESSIONAL SERVICES COMPANY
FIRST EDITION, OCTOBER 1977*

*SOFTWARE ENGINEERING A PRACTITIONERS APPROACH
ROGER S. PRESSMAN
MCGRAW-HILL 1988*

COMPILER DESIGN AND CONSTRUCTION
ARTHUR B PYSTER, PH. D.
PWS PUBLISHERS EDITION

EDI CONCEPTOS
IBM
FEBRERO 1992

TURBO PASCAL 5.5
THE COMPLETE REFERENCE
STEPHEN K. O'BRIEN
BORLAND-OSBORNE/MC GRAW HILL

APRENDIENDO Y USANDO EL DBASE III PLUS, VOLUMEN I, II
ASHTON TATE 1989

USING VENTURA PUBLISHER
DIANE BURNS
S. VENIT
LINDA J. MERCER
QUE CORPORATION EDITION
2ND. EDITION

LASER JET IIIIP PRINTER
USER'S MANUAL
HEWLETT PACKARD COMPANY
FIRST EDITION, MAY 1991

Sistema Traductor de Intercambio Electrónico de Datos

P

| | |
|---------------------|------|
| Password | 3-46 |
| Patrones de trabajo | 1-7 |
| Procesos | |
| Características | 2-33 |
| Descripción | 2-33 |
| Protocolo | 2-17 |

R

| | |
|--|------|
| Recepción de Información de la Compañía | 2-27 |
| EDI | 2-21 |
| Red de comunicaciones | 1-5 |
| Requerimientos | 2-19 |
| Funcionales | 2-19 |
| No Funcionales | 2-20 |

S

| | |
|------------------------------|------------|
| Segmento Dato | 2-17, 2-19 |
| Segmento de datos | 1-6 |
| Sesión de comunicaciones EDI | 1-7 |
| Sistemas expertos | 1-12 |
| Software de comunicaciones | 1-5 |
| Software de traducción EDI | 1-5 |

T

| | |
|---------------------------|----------------------------|
| Tareas administrativas | 1-12 |
| Traductor de Formatos | 2-21 - 2-22, 2-27, 2-29 |
| Transacción | 2-17 |
| Conjunto de | 2-19 |
| Transferencia electrónica | 1-9 |

U

| | |
|-------------------------|------|
| Unidades de información | 1-6 |
| Usuario | 2-16 |
| Directo | 2-16 |
| Indirecto | 2-16 |

V

| | |
|-----------------------------|------|
| Vendedores de productos EDI | 3-47 |
| Ventajas de EDI | 1-2 |