

**UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO**

**FACULTAD DE ESTUDIOS SUPERIORES  
CUAUTITLAN**

**SISTEMAS DE INFORMACION**

**"ANALISIS Y DESARROLLO DE PROCESOS SISTEMATIZADOS PARA UN  
PROYECTO DE ENVIO DE DINERO A DOMICILIO DE E.U. A MEXICO"**

298013

**TRABAJO DE SEMINARIO  
QUE PARA OBTENER EL TITULO DE  
LICENCIADO EN INFORMATICA**

**PRESENTA:**

**MANOLO ARIAS CERVANTES**

**ASESOR: ING. MIGUEL ALVAREZ PASAYE.**



Universidad Nacional  
Autónoma de México



**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 esta 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.

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN  
UNIDAD DE LA ADMINISTRACION ESCOLAR  
DEPARTAMENTO DE EXAMENES PROFESIONALES



UNIVERSIDAD NACIONAL  
AVENIDA DE  
MEXICO

UNIVERSIDAD NACIONAL  
FACULTAD DE ESTUDIOS  
SUPERIORES CUAUTITLÁN



DEPARTAMENTO DE  
EXAMENES PROFESIONALES

DR. JUAN ANTONIO MONTARAZ CRESPO  
DIRECTOR DE LA FES CUAUTITLÁN  
PRESENTE

ATN: Q. Ma. del Carmen García Mijares  
Jefe del Departamento de Exámenes  
Profesionales de la FES Cuautitlán

Con base en el art. 51 del Reglamento de Exámenes Profesionales de la FES-Cuautitlán, nos permitimos comunicar a usted que revisamos el Trabajo de Seminario  
Sistemas de Información: "Análisis y Desarrollo de Procesos Sistematizados  
para un Proyecto de envío de dinero a domicilio de E.U. a México".

que presenta el pasante: Arias Cervantes Manolo  
con número de cuenta: 9001062-8 para obtener el título de  
Licenciado en Informática.

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el  
EXÁMEN PROFESIONAL correspondiente, otorgamos nuestro VISTO BUENO

ATENTAMENTE

"POR MI RAZA HABLARA EL ESPIRITU"

Cuautitlán Izcalli, Méx. a 6 de Septiembre de 2001

MODULO	PROFESOR	FIRMA
<u>I</u>	<u>M.C.C. Araceli Nivón Zaghi</u>	<u>[Firma]</u>
<u>II</u>	<u>M.C.C. Valentín Roldán Vázquez</u>	<u>[Firma]</u>
<u>III</u>	<u>ING. Miguel Alvarez Pasaye</u>	<u>[Firma]</u>

## **AGRADECIMIENTOS**

A Dios, por darme vida, fé y la convicción para llevar una vida plena y llena de amor; y el logro del presente objetivo es parte importante de la vida que él me ha dictado.

A mis padres, por su gran amor, apoyo incondicional y comprensión no solo les doy las más amorosas gracias, sino que les dedico este logro, que también es suyo.

A mis hermanos, que siempre han estado a mi lado, impulsándome y brindándome toda su confianza y cariño.

A mi esposa, por haber llegado a mi vida llenándola de amor y convirtiéndome en un hombre con nuevas metas enfocadas a ella y nuestra nueva familia.

A la Universidad Nacional Autónoma de México, por brindarme una formación y ética profesional incomparables.

A cada uno de mis profesores por compartir conmigo y mis compañeros todo el conocimiento adquirido.

A todas las personas allegadas, por estar a mi lado en cada momento importante de mi vida, brindándome apoyo y una amistad invaluable.

## **OBJETIVO GENERAL.**

Analizar las características del proyecto para desarrollar e implementar los procesos sistematizados necesarios que aseguren el correcto flujo de información entre Unión del Oeste, cuyo giro es el envío de dinero en E.U.A. y MultiPaquetería, dedicada a la mensajería y paquetería en México, en una alianza comercial que pretende introducir el servicio de envío de dinero desde E.U.A. hacia México, pero con entrega a domicilio, utilizando en la mayor medida posible la infraestructura operativa y de sistemas utilizada actualmente en la empresa MultiPaquetería.

## **OBJETIVOS PARTICULARES.**

- Determinar sobre la base de los requerimientos de información de las empresas involucradas, el formato y tipo de datos que deberán transferirse entre las mismas.
- Tomando como base los procesos operativos de MultiPaquetería, determinar la frecuencia con la que se deberán establecer los procesos de transmisión de información.
- Definir los canales de comunicación y la seguridad que se implementará en las mencionadas transmisiones.
- Utilizar un servidor con una base de datos para procesar la información y recibir y enviar información buscando el mayor grado de automatización posible.
- Establecer procesos con la capacidad de poder destinar la información hacia el punto logístico más conveniente para agilizar la entrega de los envíos, tomando como base su destino final.
- Implementar en la base de datos y en los procesos desarrollados los elementos necesarios para considerar todos los puntos operativos y las políticas de negocios que el proyecto implica (validación de datos, conectividad a bases de datos operativas de MultiPaquetería, previsión de errores, consideraciones operativas, etc.).

## **HIPOTESIS.**

Con el desarrollo del presente trabajo de seminario y la implementación de los procesos automatizados y aspectos técnicos que en el mismo se describen, se logrará establecer un flujo de información conveniente para los requerimientos informativos de las empresas que participan en esta alianza comercial, utilizando en la mayor medida los actuales procesos y recursos operativos así como de sistemas con los que cuenta MultiPaquetería. De esta manera MultiPaquetería logrará el éxito del proyecto y también cumplirá con las exigencias de información que imperan en el mercado de la Mensajería y la Paquetería.

## INTRODUCCION.

Como consecuencia de las crecientes necesidades de información que giran en torno a la empresa de la Mensajería y la Paquetería en México y en el mundo, resulta imperante contar con una infraestructura tecnológica que permita cubrir dicha demanda informativa, esta ha sido una de las tareas a las que se ha dedicado MultiPaquetería desde hace algunos años, obteniendo aplicaciones administrativas y operativas que trabajan sobre bases de datos centralizadas en las que se hace disponible la información para cada punto de venta, operativo o administrativo de la empresa sin importar en que ubicación geográfica se encuentre. Obviamente esta tarea ha requerido de una gran inversión en todos los aspectos (económico, de tiempo, humano, etc.), pero ya se comienzan a ver los beneficios y frutos de ese arduo trabajo.

Como caso específico de esos frutos se tiene el proyecto que abarca el presente trabajo, un proyecto en el que gracias a la infraestructura lograda por MultiPaquetería, una empresa como Unión del Oeste en E.U.A. ha puesto sus ojos en ella para establecer una alianza comercial que ofrece un nuevo servicio para hacer llegar dinero desde E.U.A. a México con la característica de entregarlo directamente en el domicilio de los beneficiarios.

En toda alianza comercial se requiere que las partes involucradas cuenten con la información que les permita llevar un óptimo control administrativo, operativo,

financiero, etc. Como consecuencia de este requerimiento es que ha surgido este proyecto, el cual se centrará en los aspectos sistematizados vistos desde la perspectiva de MultiPaquetería, sin dejar de considerar los aspectos operativos y las políticas de negocios que giran alrededor del mismo y se buscará utilizar las herramientas tecnológicas que faciliten y hagan transparentes las interfaces entre este proyecto y la base de datos centralizada de MultiPaquetería.

En el primer capítulo del presente trabajo se hace una descripción de las características generales de las empresas que establecen la alianza comercial que hemos mencionado.

En el segundo capítulo se describe el modo y los medios que permitirán establecer un canal de comunicación entre ambas empresas.

En el tercer capítulo se describen y analizan los procesos que deberá hacer MultiPaquetería para llegar al cumplimiento del objetivo principal del trabajo.

El último capítulo describe los recursos que se utilizarán, así como el desarrollo de los procesos desde el punto de vista de Sistemas de Información dado el análisis realizado en el capítulo anterior.

## INDICE.

<b>TITULO</b>	<b>PAG.</b>
<b>Objetivo General.</b>	
<b>Objetivos Particulares.</b>	
<b>Hipótesis.</b>	
<b>Introducción.</b>	
<b>Indice.</b>	
<b>Capítulo 1.</b>	
<b>Características comerciales y tecnológicas de las empresas involucradas.</b>	
1.1 Unión del Oeste.	13
1.1.1 Servicios.	13
1.1.2 Servicios en México.	14
1.1.3 Tecnología.	15
1.2 MultiPaquetería.	17
1.2.1 Alianzas estratégicas.	18
1.2.2 Certificación ISO 9001 en MultiPaquetería.	19
1.2.3 Servicios.	21
1.2.4 Tecnología.	23
<b>Capítulo 2.</b>	
<b>Comunicación entre las empresas para el Proyecto.</b>	
1.3 Dispositivos de comunicación.	26
1.3.1 Módems.	27
1.3.1.1 Limitaciones de velocidad.	28

D. Proceso de Cancelaciones.	124
E. Proceso para Informar situaciones.	126

**Referencias electrónicas**

**Bibliografía.**

1.3.1.2	Los comandos Hayes.	34
1.3.2	Ruteadores	34
1.3.2.1	Donde utilizar un ruteador	36
2.2	Esquema de Comunicación.	37
2.3	Información a transmitir y seguridad de la misma.	39
2.3.1	Información de Unión del Oeste a MultiPaquetería.	40
2.3.1.1	Frecuencia de transmisión.	41
2.3.1.2	Formato de la Información.	42
2.3.2	Información de MultiPaquetería hacia Unión del Oeste.	45
2.3.2.1	Frecuencia de transmisión.	48
2.3.2.2	Formato de la Información.	48
2.3.3	Mecanismos de seguridad de la información.	50

### **Capítulo 3.**

#### **Análisis de procesos a realizar por parte de MultiPaquetería.**

3.1	Transmisión de información hacia regiones de MultiPaquetería.	53
3.1.1	Cobertura asignada al proyecto.	53
3.1.2	Determinación de región destino.	55
3.1.3	FTP como medio de transmisión.	57
3.2	Integración de envíos a Base de Datos Centralizada de MultiPaquetería.	60
3.2.1	Carga Masiva.	60
3.2.2	Impresión de cheques y generación de paquetes.	63
3.3	Procesos de cancelaciones.	66
3.3.1	Cancelaciones por vigencia.	67

3.3.2 Cancelaciones por solicitud de Unión del Oeste.	67
3.4 Actualización de situaciones de envíos en MultiPaquetería.	69
3.5 Notificar status de envíos a Unión del Oeste.	69
3.5.1 Obtención e interpretación de Información	69
3.5.2 Causas de no entrega.	71
3.5.3 Generación y Transmisión de Información.	72

## **Capítulo 4.**

### **Desarrollo Técnico del Proyecto.**

4.1 Software y servidor del proyecto	75
4.2 Características importantes del software utilizado.	77
4.2.1 Nt.	77
4.2.2 View Now.	78
4.2.3 Oracle 8.1.5.0.0.	78
4.2.4 Plsql y Utl_file.	81
4.3 Definición y desarrollo de procesos.	82
4.3.1 Importación y distribución de información.	83
4.3.2 Comunicación con base de datos central.	95
4.3.3 Proceso de cancelaciones.	97
4.3.4 Informe de situaciones a Unión del Oeste.	100

### **Conclusiones.**

### **Apéndice**

A. Sobre la creación de la base de datos.	111
B. Creación de tablas del usuario.	114
C. Proceso de incorporación y distribución de Información.	116

**CAPITULO 1. CARACTERISTICAS COMERCIALES Y  
TECNOLOGICAS DE LAS EMPRESAS INVOLUCRADAS.**

## **1.1 UNION DEL OESTE.**

Unión del Oeste es una empresa norteamericana que pertenece al grupo empresarial First Data y ofrece el servicio más rápido de envío y recepción de dinero en el ámbito mundial. Es una empresa que ofrece sus servicios en los cinco continentes a través de mas de 80,000 puntos de venta ubicados en más de 170 países.

### **1.1.1 SERVICIOS.**

Como giro y principal servicio que ofrece Unión del Oeste podemos mencionar:

Envío de dinero.- Este servicio permite a clientes y negocios el envío de dinero a través de cualquiera de sus puntos de venta. Este servicio tarda como máximo 24 hrs. desde el momento de la contratación del servicio, salvo casos muy especiales debido a la situación física del destino del servicio.

El costo del servicio depende de diversos factores tales como el servicio seleccionado, el origen del servicio, el destino del servicio, el monto de dinero que se pretende enviar, etc.

El monto máximo de dinero que se puede enviar depende en gran parte del destino del servicio y de las leyes que imperan en esa localidad, así como de los

lineamientos que se hayan estipulado en los convenios comerciales establecidos en esas localidades, ya que en muchos de los casos la entrega del servicio se realiza en establecimientos que no son propiedad de Unión del Oeste y esas entregas se rigen por convenios comerciales.

El importe de dinero que se entrega en el servicio se paga en la divisa del país en donde está la ubicación destino del servicio, aunque en algunas localidades se ha especificado que el pago sea siempre en dólares americanos.

Unión del Oeste ofrece también sus servicios a través de una página en Internet, obviamente los servicios a través de Internet están limitados en comparación a los que se pueden realizar en los puntos de venta.

### 1.1.2 SERVICIOS EN MEXICO.

En México, Unión del Oeste cuenta con el servicio de "Dinero en Segundos" que se basa en alianzas comerciales firmadas con empresas mexicanas tales como Elektra, Bitel y Centrales de Autobuses. Otro servicio que se presta es utilizando giros telegráficos a través de Telecomm. En todos los casos el dinero es recogido en los establecimientos de estas empresas.

Como proyecto nuevo para esta empresa transnacional está el de poder hacer la entrega del dinero al beneficiario del servicio directamente en su domicilio, para lo

cual fue necesario que Unión del Oeste buscara una empresa que contara con la infraestructura necesaria para contar con este cometido, inclinándose por MultiPaquetería debido precisamente a su infraestructura, la cual permitió ofrecer a Unión del Oeste la cobertura y la capacidad operativa que se requiere para este proyecto, y es precisamente esta nueva tendencia de servicio la que motiva el presente trabajo de seminario.

### 1.1.3 TECNOLOGÍA.

Por razones obvias de seguridad, Unión del Oeste no ha proporcionado toda la información con respecto a la infraestructura tecnológica y de telecomunicaciones con la que cuenta, pero es evidente que cuenta con los elementos necesarios para hacer llegar información de manera eficiente a casi cualquier parte del mundo, sin utilizar totalmente por los canales públicos de Internet. Utiliza la capacidad de transmisión de la fibra de vidrio y grandes equipos Tandem como concentradores y transmisores de información.

Básicamente tienen puntos de venta en línea con estos equipos Tandem, los cuales tienen bases de datos en donde se guardan las órdenes de servicio y sus características, estas bases de datos son utilizadas por procesos que se encargan de distinguir el destino de la información y hacerla llegar al mismo, a través de procesos preestablecidos, utilizando en la mayoría de los casos canales de comunicación dedicados hacia las empresas con las que tiene alianzas comerciales.

Como se ha mencionado, Unión del Oeste cuenta también con un portal en Internet en el cual se pueden contratar algunos de los servicios que ofrece, y también se brinda información en general sobre la empresa, sus servicios, los mecanismos de contratación de servicios, ubicación de puntos de venta, etc.

## **1.2 MULTIPAQUETERIA.**

MultiPaquetería, empresa 100% mexicana, está dedicada al transporte de mensajería y paquetería, así como al giro de la operación logística, distribución y aprovisionamiento.

Entendemos como transporte al traslado por cuenta propia o de terceros de mercancía y objetos de un origen a uno o varios destinos. Como mensajería, el traslado de sobres de hasta un kilogramo con tiempos preestablecidos.

Consideramos como paquetería al traslado de paquetes con tiempos preestablecidos y condiciones de naturaleza, peso y volumen; y, como operación logística de distribución y aprovisionamiento, el almacenaje y traslado de mercancías de un cliente hasta un destino, acoplando las necesidades y participando en el proceso administrativo y comercial del cliente.

MultiPaquetería es una empresa del Grupo A.D.O., que posee una infraestructura de 4500 unidades de transporte, 35 Centros de Distribución, 300 destinos distribuidos en 58 cabeceras con 22 regiones para cubrir el territorio nacional.

Su servicio de mensajería y paquetería inicia en 1939, cuando la línea de Autobuses de Oriente inauguró sus rutas por el sudeste de la República. Desde entonces, hemos desplazado documentos y paquetes para beneficio de nuestros clientes. Más

de 50 años por los caminos de México han dado a MultiPaquetería un alto nivel de madurez para garantizar el cumplimiento de los compromisos de puntualidad y seguridad adquiridos con los clientes.

En MultiPaquetería, el Sistema de Calidad está sustentado en un Modelo de Calidad, mismo que ha permitido convertirla en una organización totalmente orientada al cliente. Al igual que su infraestructura y su experiencia, el cumplimiento de dicho Modelo de Calidad es permanente y su enfoque de Calidad esta dirigido a darle al cliente lo que solicita.

#### 1.2.1 ALIANZAS ESTRATEGICAS.

MultiPaquetería, en su afán y dedicación de otorgar los mejores servicios de mensajería y paquetería durante 60 años, amplía su cobertura mundial realizando alianzas con las más importantes empresas internacionales en mensajería, transportación y carga. De esta forma aumenta la gama de servicios con un solo objetivo: Satisfacer las necesidades de nuestros clientes:

Entre estas alianzas, podemos mencionar las establecidas con empresas tales como:

- ABF
- BExpress
- GreyHound

- Federal Express
- TNT
- Iberia
- Mexicana de Aviación
- Entre otras.

## 1.2.2 CERTIFICACION ISO 9001 EN MULTIPAQUETERIA

Nuestro modelo de Calidad demostró una vez más ser la principal herramienta de trabajo y base de la consolidación del sistema de calidad de MultiPaquetería. Del 8 al 17 de febrero del año en curso, se auditaron varias regiones por parte del grupo certificador respecto a la Norma ISO 9001, obteniendo resultados satisfactorios en los conceptos de mantenimiento y de certificación.

Las auditorías de mantenimiento tienen como objetivo, verificar que se cumplan los 20 requisitos de la norma ISO 9001, los cuales aseguran que MultiPaquetería mantiene una Sistema de Calidad como medio para asegurar la satisfacción total del cliente.

Estas auditorías estuvieron dirigidas a las zonas de: Ciudad del Carmen, Campeche, Mérida, Cancún, Chetumal, Orizaba, Córdoba, Veracruz, Jalapa, Villa Hermosa, Cárdenas, Coatzacoalcos, Minatitlán, Oaxaca, Juchitán, Aguascalientes, León,

Morelia, Querétaro, Tijuana, Hermosillo, Tuxtla Gutiérrez, Matamoros, Reynosa, Puebla y Tlaxcala; las cuales ya habían sido certificadas el 27 de febrero de 1998.

Por otro lado las auditorías de certificación tienen el propósito de hacer un análisis para determinar si el Sistema de Calidad y sus resultados cumplen con las disposiciones establecidas por la Norma ISO 9001 y si éstas son implantadas eficazmente. Los resultados alcanzados permitieron obtener, el 26 de Marzo del presente el certificado ISO 9001-NMX-CC-003 a : Chihuahua, Ciudad Juárez, Tapachula, Tampico, Poza Rica, Tehuacán, Teziutlán, Toluca, Pachuca, Acapulco, Cuernavaca y Torreón.

Es así como se usan ya 39 zonas certificadas incluyendo al Corporativo México. Mediante este hecho se confirma que MultiPaquetería cuenta con un Sistema de Calidad consolidado en el ámbito nacional. Así mismo, es un reconocimiento internacional que hace que MultiPaquetería sobresalga como la primera y única empresa de mensajería y paquetería en nuestro país certificada bajo esta Norma de Calidad mundialmente aceptada.

### 1.2.3 SERVICIOS.

Contamos con el apoyo de sucursales y centros de distribución ubicados en todo el país, con el mejor servicio de Mensajería y Paquetería a más de 300 destinos Nacionales y 500 Internacionales.

Servicio Ocorre: El destinatario debe presentarse a recoger su paquete o sobre en la sucursal donde se lo ha enviado el remitente.

Entrega a domicilio: Entrega del envío directamente en el domicilio del destinatario.

Recolección a domicilio: MultiPaquetería recoge el envío en el domicilio del cliente.

Guías Express: Compra anticipada de guías para el envío de Mensajería de hasta 1 kg. incluyendo entrega a domicilio, por el mismo precio a todos los destinos. (Compra mínima de 20 guías y descuento por volumen)

Guías MultiPaquetería: Compra anticipada de guías para el envío de Paquetería de 5 kg hasta 50 kg incluyendo entrega a domicilio. (Compra mínima de 20 guías y descuento por volumen)

Acuse de recibo: Confirmación por escrito de la entrega de su envío.

Valor declarado: Con un pago mínimo adicional su envío queda asegurado en caso de daño parcial o total del mismo por alguna contingencia.

Venta de Tarjeta Ladatel y Iusacell.

Servicio Internacional.

Multifast: Servicio adicional de mensajería a través de Telefax.

Pack Center: Es un nuevo concepto en servicios diseñados exclusivamente para integrar y atender en un mismo lugar las necesidades de empaque profesional de alta seguridad, mensajería, paquetería, oficina y comunicación que usted pudiera tener.

Condiciones del Servicio:

El tiempo de entrega es de 24 hrs. por cada 1000 km. de distancia entre el origen y el destino. (aplican restricciones).

Aceptamos envíos de hasta 80 kg. y con dimensiones máximas de 80 x 80 x 80 cm. No podemos transportar armas, valores y otros artículos de carácter peligroso o nocivo.

#### 1.2.4 TECNOLOGÍA.

La Mensajería y Paquetería es un giro que tiene grandes exigencias de comunicación y disponibilidad de información. Al ser este un giro que por su naturaleza necesita disponer de un gran número de puntos de localización en toda la república, necesita contar con una infraestructura de telecomunicaciones y sistemas que permita tener disponible la información de todos esos puntos geográficos de la manera más oportuna para poder cumplir con sus objetivos empresariales.

Debido a estas exigencias y a la evolución tecnológica que se vive día a día en todos los ámbitos empresariales, MultiPaquetería se dio a la tarea de diseñar e implantar la mencionada infraestructura, por lo que en 1997 comenzó su proyecto de telecomunicaciones, el cual consistió en implementar una intranet a escala nacional basada en fibra de vidrio y canales de comunicación dedicados que enlazan puntos logísticamente establecidos para la operación de la empresa, a través de diferentes recursos de comunicación como DS 0's, routers, switches, antenas, etc. Este proyecto se concluyó con gran éxito al grado que ha sido premiado por instituciones de la rama de las telecomunicaciones.

En 1999 se comenzó con un proyecto que buscaba la implementación de un sistema ERP, eligiendo SAP/R3 y el desarrollo de una aplicación basada en JAVA que permitiera concentrar la información operativa en una base de datos Oracle centralizada, la cual residiría en equipos SUN de gran capacidad ubicados en el

corporativo del D.F.. Este proyecto permite tener disponible la información operativa en línea, es decir, que si un cliente contrata el servicio de MultiPaquetería en una sucursal de Hermosillo (por ejemplo), inmediatamente una persona que radique en Mérida (por ejemplo), podrá consultar esa transacción y dar un seguimiento de cada movimiento que tiene ese envío hasta el momento de su entrega. Este proyecto está en fase de instalaciones finales por lo que la información contenida en esta base de datos ya es de total importancia.

MultiPaquetería cuenta con una página en Internet en la que se pueden conocer los datos trascendentes de la empresa y está desarrollándose un portal en el que además de toda esta información, se podrán contratar servicios en línea y rastrear envíos por diferentes criterios de búsqueda.

Las características tecnológicas, las soluciones logísticas y las políticas de calidad aquí mencionadas han colocado a MultiPaquetería dentro de las mejores empresas de Mensajería y Paquetería en México, ya que son elementos que permiten que MultiPaquetería trabaje bajo la consigna de ofrecer a sus clientes los servicios y la información que ellos requieren y con la calidad requerida.

**CAPITULO 2. COMUNICACION ENTRE LAS EMPRESAS PARA EL  
PROYECTO.**

## 2.1 DISPOSITIVOS DE COMUNICACION.

Para este proyecto de envíos de dinero a domicilio es necesario establecer un canal de comunicación entre Unión del Oeste y MultiPaquetería que permita la transmisión de la información generada por Unión del Oeste<sup>1</sup>, y por otro lado también la información generada por MultiPaquetería<sup>2</sup>.

Para este fin se recurre a la utilización de dispositivos de comunicación que permitan comunicación entre un equipo Tandem designado por Unión del Oeste como servidor del proyecto, y un equipo con Windows NT Server, designado por MultiPaquetería como servidor para el proyecto. Como ya se ha mencionado, el desarrollo del presente trabajo se hace desde la perspectiva de MultiPaquetería, por lo que será precisamente la parte tecnológica de esta empresa la que se detallará en su oportunidad.

Por cuestiones de seguridad y de confidencialidad, ambas empresas han determinado que no se expliquen a detalle los puntos finos sobre esta comunicación, por lo que se limitará a comentar el proceso de comunicación y los dispositivos utilizados.

---

<sup>1</sup> Nuevas órdenes de servicio, así como solicitudes de cancelación de servicios.

<sup>2</sup> Acerca de las situaciones de los servicios de los envíos en las bases de datos operativas de MultiPaquetería.

### 2.1.1 MODEMS.

Cuando queremos comunicar dos PCs, esto es, intercambiar datos entre unos y otros, del mismo modo que nosotros podemos hablar con nuestros vecinos, se hace necesario algo que una a los dos ordenadores, es decir: un canal. En el caso humano, no hay problema si utilizamos la voz. Del mismo modo, con un cable que se conecte a los puertos en serie de ambas máquinas, y un buen programa todo resuelto.

Si lo que queremos es hablar con el vecino de la casa de enfrente, a gritos nos entenderemos mal, y perderemos gran parte de lo que nos digamos. Con los ordenadores, tres cuartos de lo mismo: la longitud del cable no puede exceder cierto límite, porque aparecen problemas con el redireccionamiento, pérdida de potencia, etc.

Nosotros, no nos quedamos sin hablar con el vecino: agarramos el teléfono solucionamos el problema. Para los ordenadores podemos utilizar la RTC o Red Telefónica Conmutada.

Ahora, para el ordenador es imprescindible el uso de un módem para que "pueda hablar" por teléfono. El módem se ha convertido hoy en día en un periférico muy necesario, imprescindible a la hora de compartir datos a larga distancia. Ya sea interno o externo, más o menos rápido, con características de fax o sin ellas, nos

permitirá que nuestro PC acceda a todo un mundo plagado de servicios telemáticos de gran valor.

Para aquel que no lo sepa, módem es la abreviatura de MODulador DEModulador, es decir, un aparato que codifica (MODula) los datos que quiere mandar el ordenador por teléfono, en señales; y las envía por la línea. Cuando son recibidas al otro lado, el módem de allí los DEModula, es decir, los decodifica tomándolas en datos de ordenador.

Naturalmente, el módem es también el encargado de "marcar" los números de teléfono a los que se quiere llamar.

#### **2.1.1.1 Limitaciones de velocidad.**

Una de las cosas que va a limitar grandemente las comunicaciones por línea telefónica, es la calidad de sonido que se puede obtener.

Cuando Alexander Graham Bell inventó el teléfono, supo que la voz no era oída, ni mucho menos, igual que salía de la voz del hablante. Esto se debe a que el rango de frecuencias permitido por la línea es realmente pequeño, de unos pocos miles de hertzios. Supongo que esto a muchos les estará sonando a chino, de modo que me explico:

El sonido, físicamente hablando, no es más que una vibración, una onda, es decir, una transmisión de energía de unos cuerpos a otros sin que haya entre estos intercambio de materia (ya sé que esto "suena" un poco raro, pero es así). Si la onda tiene una frecuencia baja, es decir, si la onda vibra pocas veces por segundo, tenemos un sonido grave. Si por el contrario, es de una frecuencia alta (muchas vibraciones por segundo) lo percibimos como un sonido agudo. El oído humano, normalmente, no será capaz de oír sonidos cuyas frecuencias no se encuentren entre los 20 Hz y los 18000 Hz, es decir, que el sonido más grave que percibiremos será aquel cuya onda haga vibrar la materia veinte veces por segundo; y el más agudo que percibiremos será aquel que haga vibrar la materia 18000 veces por segundo (estos números son aproximados, no todo el mundo oye esto). Ahora empieza lo bueno:

Comprender como funciona el sonido digital respecto del analógico, puede ayudar a entender muchas cosas acerca de la transmisión de datos.

Para empezar, el sonido que nosotros oímos es analógico, porque, tomando cualquier instante, nosotros podemos encontrar sonido. Un CD, graba en sonido digital; y un sonido se considera digital cuando, al contrario que el analógico, tomando un instante cualquiera, no podemos afirmar que vayamos a encontrar sonido. Es como si sonara un poquito, se callara, sonara otro poco ... y diera la impresión de que el sonido no se corta ... Por ello, para escuchar la música de un CD, hace falta un DAC (Digital Analogic Conversor) es decir, algo que nos permita convertir el sonido digital en analógico y viceversa.

En un CD grabamos por ejemplo música. El CD cubre un rango de frecuencias desde 0 hasta 44000 Hz, es decir, el doble del rango de audición humana (porque los que hicieron el CD contaron con que oíamos desde 0 hasta 22000 Hz para redondear incluso con la gente de oído más fino).

El rango de muestreo necesita ser el doble que la mayor de las frecuencias que quieran ser registradas. Se notará que una cinta no se oye igual que un CD. Esto es, además de porque al grabar en una cinta solemos captar ruido, porque el rango que alcanza la cinta es de 0 a 11000-22000 Hz (normalmente y según la calidad de la cinta). Por teléfono, este rango empieza algo más arriba de cero, y según línea y aparatos, llega hasta los 5000 u 11000 Hz (y encima es sonido monoaural, es decir, que sólo hay un canal de sonido) el sonido estereofónico que tiene dos canales suena mejor porque imita a nuestras DOS orejas. Como podemos observar, la calidad de sonido del teléfono, deja bastante que desear.

La línea telefónica, no nos permite tampoco hacer los cambios de frecuencia que queramos; no nos permite más que cambiar la frecuencia del sonido que emitimos más que unas pocas veces por segundo (a esto se le llama ancho de banda).

Ahora, ya estamos preparados para saber como va esto del módem. El ordenador "uno" manda al ordenador "dos" un byte. Un byte, como todos sabremos, está codificado por un octeto de bits. Un bit puede ser un cero o un uno. Por ejemplo, supongamos que queremos mandar el siguiente byte a otro ordenador:

10101010

Para ello, hacemos lo siguiente:

Lo primero es establecer un convenio.

Por ej: Si se manda una señal a 1000 Hz es un 0.

Si se manda una señal a 3000 Hz es un 1.

Para ello, enviamos sonido en la frecuencia del 1, enviamos sonido en la otra frecuencia del 0, enviamos sonido en la frecuencia del 1... esto es "grosso modo" lo que haría el módem: enviar los bits codificados en forma de sonidos de frecuencias determinadas. Como la RTC tiene un ancho de banda de 2400, es decir, podemos realizar 2400 cambios de frecuencia por segundo, así que la velocidad máxima a la que podremos enviar será de 2400 bits por segundo, es decir 2400 baudios.

Supongamos que en vez de utilizar una frecuencia para cada bit, utilizamos una frecuencia para cada dos bits, 00 01 10 11 Necesitaríamos cuatro frecuencias distintas, pero estaríamos enviando el doble de bits en cada señal, es decir, podríamos transmitir a 4800 baudios. ¿ Entonces no podríamos utilizar ocho millones

de frecuencias para enviar mas bits por cada frecuencia distinta y así poder mandar una pila de bits por cada señal y tener un módem de un mayor número de baudios?.

Actualmente, lo más que se ha llegado a meter son doce bits por señal. Con un sencillo cálculo sabremos que estamos utilizando  $2^{12} = 4096$  frecuencias diferentes, y transmitimos a 28800 baudios. Nótese como lo de la frecuencia tenía su razón de ser. Si quisiéramos meter más frecuencias, para el teléfono no habría ninguna diferencia entre las que estuviesen más cercanas. De modo que al módem le daría lo mismo una combinación de bits que otra.

Aún así, hay varios modos de modulación (es decir, el modo en que a un bit o a una combinación de estos se le asigna una frecuencia), como pueden ser PSK o DSPK.

Aunque enviando 4096 frecuencias distintas, en la línea telefónica están tan próximas (hay tan poca diferencia entre unas y otras) que el más mínimo ruido puede hacer que una señal se transforme en otra. Naturalmente esto provocará que los datos que recibimos ya no sean los mismos que los que fueron enviados; por eso hay que tener cuidado con las líneas ruidosas.

Cuando oímos que algo está grabado a 44000 Hz, es porque se han tomado en un segundo, 44000 muestras de ese sonido (es decir, al convertir el sonido real (analógico) a sonido digital, se ha hecho a razón de 44000 muestras de sonido digital por cada segundo de sonido analógico). Para representar la amplitud de cada muestra, podemos utilizar números de 8 y 16 bits.

Un número de 8 bits nos permite alcanzar rangos de 0 a 255, mientras que uno de 16 bits nos permite rangos de 0 a 65535.

Como podemos observar, cuantas más muestras de un sonido tengamos, y cuanto mejor podamos decir cuanto mide cada una de esas muestras, más se parecerá el sonido al original.

Y es que los he engañado un poco al principio con eso de los Hz. Lo que realmente nos da el rango de frecuencias va a ser el número de bits que usemos: 8 ó 16. Ahora bien, cuantos más Hz, es decir, más muestras de sonido por segundo tengamos, mejor oiremos.

Un CD está grabado a 16 bits (que permite definir la frecuencia de cada muestra dentro de un rango de 0 a 65535), y 44000 Hz (tomamos 44000 muestras de sonido por segundo). Esto es más que suficiente para el oído humano.

Para representar una imagen, ponemos juntos muchos puntos pequeñitos, que en conjunto forman una imagen. Ahora, ¿Porque se ve una imagen? Porque hay muchos puntos pequeños en poco sitio. Al acercar la lupa, hemos hecho de alguna forma, que los puntos crezcan y que el espacio sea mayor. Por eso ya no vemos una imagen. Bien. Algo parecido ocurre con el sonido. El sonido es una onda, y al grabarlo en el CD lo que grabamos es una descripción de la onda. Es decir, que tomamos un segundo de onda, lo dividimos en un montón de trozos, y de cada trozo,

nos quedamos con la parte más representativa (el punto); y muchos puntos juntos hacen algo que es muy parecido a la onda original (pero no igual).

Entonces... ¿ por qué se utiliza el sonido digital en vez del analógico, si el sonido digital no es igual al original y, en cambio, el sonido analógico si lo es?

La respuesta es bastante simple. Un disco de vinilo graba el sonido analógicamente, grabando las vibraciones del sonido como un microsurco en una pista y, al reproducir, los microsurcos de la pista hacen que la aguja del tocadiscos vibre del mismo modo que lo hizo la onda de sonido original. Las vibraciones de la aguja son amplificadas.

Ahora, ¿ qué se oye mejor, un CD o un disco de vinilo ? Ah ! Eso es porque en la grabación de un CD no hay posibilidad de que entren ruidos porque lo que se graban son dígitos (ceros y unos). Además, en el caso del disco de vinilo, hay contacto físico entre la aguja y el disco; mas no lo hay entre el cabezal del lector CD y el disco.

### **2.1.1.2 Los comandos Hayes.**

Los comandos Hayes son necesarios para controlar el módem ya que las señales del puerto en serie no son siempre suficientes para indicar al módem tareas tales como el número a marcar

Los principales comandos Hayes son:

- AT: Alertar al módem: aviso de orden
- ATH0 y ATH1: para colgar y descolgar.
- ATL3: Fijar el volumen del altavoz al máximo
- ATM0: Fijar el volumen del altavoz al (volumen 0)
- ATA: responde a una llamada
- ATSX=YY: para fijar el contenido del registro X con al valor YY; ATDTXYZ y ATDPXYZ: para llamar respectivamente por tonos o por pulsos al número XYZ
- AT&V: Mostrar configuración interna
- ATZ: Para inicializar el módem.

### 2.1.2 RUTEADORES.

Un ruteador es un dispositivo de propósito general diseñado para segmentar la red, con la idea de limitar tráfico de broadcast y proporcionar seguridad, control y redundancia entre dominios individuales de broadcast, también puede dar servicio de firewall y un acceso económico a una WAN.

El ruteador opera en la capa 3 del modelo OSI y tiene más facilidades de software que un switch. Al funcionar en una capa mayor que la del switch, el ruteador distingue entre los diferentes protocolos de red, tales como IP, IPX, AppleTalk o

DECnet. Esto le permite hacer una decisión más inteligente que al switch, al momento de reenviar los paquetes.

El ruteador realiza dos funciones básicas:

El ruteador es responsable de crear y mantener tablas de ruteo para cada capa de protocolo de red, estas tablas son creadas ya sea estáticamente o dinámicamente. De esta manera el ruteador extrae de la capa de red la dirección destino y realiza una decisión de envío basado sobre el contenido de la especificación del protocolo en la tabla de ruteo.

La inteligencia de un ruteador permite seleccionar la mejor ruta, basándose sobre diversos factores, más que por la dirección MAC destino. Estos factores pueden incluir la cuenta de saltos, velocidad de la línea, costo de transmisión, retraso y condiciones de tráfico. La desventaja es que el proceso adicional de procesamiento de frames por un ruteador puede incrementar el tiempo de espera o reducir el desempeño del ruteador cuando se compara con una simple arquitectura de switch.

#### **2.1.2.1 Donde utilizar un ruteador.**

Las funciones primarias de un ruteador son:

- Segmentar la red dentro de dominios individuales de broadcast.
- Suministrar un envío inteligente de paquetes.

- Soportar rutas redundantes en la red.

Aislar el tráfico de la red ayuda a diagnosticar problemas, puesto que cada puerto del ruteador es una subred separada, el tráfico de los broadcast no pasaran a través del ruteador.

Otros importantes beneficios del ruteador son:

- Proporcionar seguridad a través de sofisticados filtros de paquetes, en ambiente LAN y WAN.
- Consolidar el legado de las redes de mainframe IBM, con redes basadas en PCs a través del uso de Data Link Switching (DLSw).
- Permitir diseñar redes jerárquicas, que delegen autoridad y puedan forzar el manejo local de regiones separadas de redes internas.
- Integrar diferentes tecnologías de enlace de datos, tales como Ethernet, Fast Ethernet, Token Ring, FDDI y ATM.

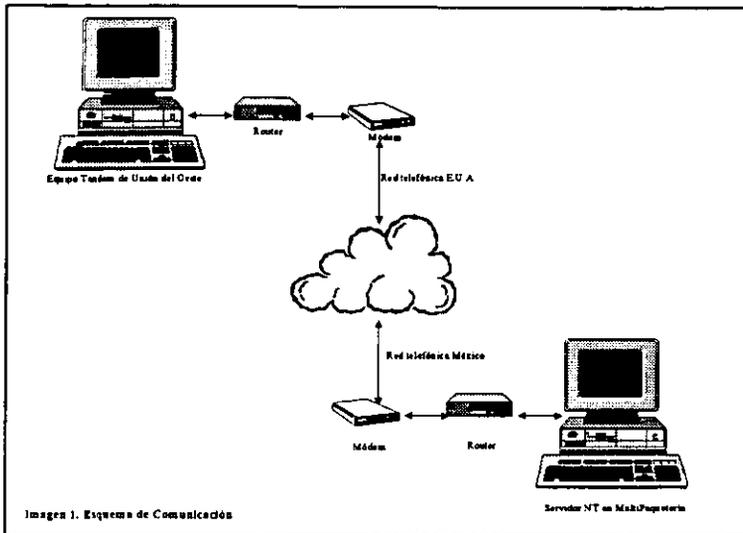
## **2.2 ESQUEMA DE COMUNICACION.**

Como se ha mencionado, por cuestiones de seguridad no se detallará la configuración de los dispositivos, pero el modo de establecer la comunicación entre los equipos designados como servidores en el proyecto, es la siguiente:

Ambos servidores están configurados para tener como predeterminado un router cada uno, designado exclusivamente para este fin, este router tiene conectado un módem, el cual será el dispositivo que se encargará de enlazar los routers a través de una llamada telefónica hacia el número que tiene asignado el módem al que se conectará finalmente. Esta llamada telefónica se realiza de manera automática cuando uno de los servidores ejecuta un comando de red dirigido hacia la dirección IP del otro servidor del proyecto (por ejemplo un PING), por lo tanto, lo primero para establecer la comunicación es que el servidor origen de la transmisión genere el comando de red hacia el servidor destino, espere a que la llamada telefónica concrete la comunicación y posteriormente podrá generar la transmisión utilizando FTP<sup>3</sup>, al terminar la transmisión y los módems detectar un lapso de tiempo sin transmisión de datos, la llamada telefónica se cortará cortando la comunicación entre los servidores. Véase la imagen 1 para ilustrar el proceso.

---

<sup>3</sup> FTP es un Protocolo de Transmisión de Archivo (File Transfer Protocol)



### 2.3 INFORMACION A TRANSMITIR Y SEGURIDAD DE LA MISMA.

El presente proyecto de envío de dinero a domicilio exige una comunicación muy estrecha entre las empresas involucradas para poder llevar el control de cada una de las transacciones generadas y cerrar los círculos operativos e informativos del proyecto, con esta finalidad es que se presentan las características de la información que estará transmitiéndose desde Unión del Oeste hacia MultiPaquetería y viceversa, sobre la base de los datos manejados por cada una de las empresas y que son de utilidad para la otra.

Como es de imaginarse, ante la gran diferencia que hay entre los giros comerciales de ambas empresas, surgen algunas discrepancias en cuanto al modo en que se

manejan ciertos datos, discrepancias que serán consideradas en los procesos que se desarrollarán mas adelante.

### 2.3.1 INFORMACIÓN DE UNION DEL OESTE A MULTIPAQUETERIA.

Un término importante en este proyecto es **Status o tipo de registro**, que básicamente se refiere a identificar la situación de los envíos desde el momento mismo de su generación hasta su procesamiento final que puede derivarse de ser entregado al beneficiario o cancelado por diversas causas. Estos status tienen una equivalencia numérica y los status que puede recibir MultiPaquetería de Unión del Oeste son:

110. - Este tipo de registro significa que es un envío nuevo. Este status sirve para que MultiPaquetería identifique una nueva transacción y proceda con el procesamiento de la misma buscando entregarla al beneficiario.

111. - Este tipo de registro significa que Unión del Oeste está solicitando a MultiPaquetería la cancelación de la transacción en cuestión por causas imputables a Unión del Oeste y su operación. Obviamente las transacciones que se están tratando de cancelar, son transacciones que anteriormente fueron transmitidas con un status 110.

120. - Este identificador de registro es clasificado como el registro 'trailer' y su finalidad es la de validar los registros que vienen en cada archivo transmitido.

### **2.3.1.1 Frecuencia de transmisión**

Basándose en las horas de mayor afluencia en la contratación de los servicios de envío de dinero y buscando integrar de la manera más eficiente los envíos a la operación de MultiPaquetería se han estipulado los siguientes horarios para transmitir la información desde Unión del Oeste hacia MultiPaquetería:

Primera transmisión a las 5:00 hrs., la cual contendrá los nuevos envíos y las solicitudes de cancelación de las transacciones generadas el día anterior de las 18:00 hrs. a la hora de cierre de los diferentes puntos de venta de Unión del Oeste.

Segunda transmisión a las 11:00 hrs., que contendrá la información de los nuevos envíos y las solicitudes de cancelación de las transacciones generadas en el día desde la hora de apertura hasta las 11:00 hrs. en los diferentes puntos de venta de Unión del Oeste.

Tercera transmisión a las 18:00 hrs., que contendrá la información de los nuevos envíos y las solicitudes de cancelación de las transacciones generadas en el día desde las 11:00 hrs. hasta las 18:00 en los diferentes puntos de venta de Unión del Oeste. Lógicamente este será el archivo que contendrá más información.

### 2.3.1.2 Formato de la Información

La nomenclatura que llevarán los archivos a transmitir por parte del Unión del Oeste será bajo las siguientes características:

Supongamos que la fecha de la transmisión es el día 20 de Julio del 2001, los archivos a transmitir serán:

5:00 hrs. → um072001.txt  
11:00 hrs. → um072002.txt  
18:00 hrs. → um072003.txt

En donde:

**um** .- Identifica que es un archivo transmitido de unión de Oeste a multiPaquetería.

**0720** .- Es el mes y el día de la transmisión (mdd).

**01, 02 y 03** .- Se refieren a el número de transmisión realizada al día de acuerdo a los horarios ya mencionados.

Los archivos que Unión del Oeste transmitirá a MultiPaquetería se basarán en la siguiente descripción o layout:

Para los registros de datos véase la tabla 1:

	<b>NOMBRE DEL CAMPO</b>	<b>TIPO</b>	<b>LONG</b>	<b>DEC</b>	<b>DESCRIPCION</b>
1	RECORD_TYPE	CHARACTER	3		STATUS O IDENTIFICADOR DE REG. (110 ó 111)
2	GUIA	CHARACTER	15		REFERENCIA DE MULTIPQUETERIA
3	MTCN	CHARACTER	10		REFERENCIA UNION DEL OESTE
4	TRANSTYPE	CHARACTER	3		'CHQ' = CHEQUE, 'EFE' = EFECTIVO
5	ORIGIN	CHARACTER	3		ABREVIACION DEL ESTADO ORIGEN EN E.U.A.
6	DATETIME	CHARACTER	16		FECHA DE LA TRANSMISION: FORMATO dd-mm-yyyy hh:mm
7	AMOUNTPESOS	NUMERIC	12	2	MONTO DE LA OPERACION EN PESOS
8	SENDNAME	CHARACTER	40		NOMBRE DE LA PERSONA QUE CONTRATO EL SERVICIO
9	SENDPREFIX	CHARACTER	7		LARGA DISTANCIA DEL CONTRATANTE
10	SENDPHONE	CHARACTER	7		TELEFONO DEL CONTRATANTE
11	RECNAME	CHARACTER	40		BENEFICIARIO PRINCIPAL
12	REC2NAME	CHARACTER	40		BENEFICIARIO SECUNDARIO (SOLO EN CASO DE LA AUSENCIA DEL PRINCIPAL)
13	RECSTREET	CHARACTER	40		CALLE Y NUMERO DEL BENEFICIARIO
14	RECCOLONY	CHARACTER	30		COLONIA
15	RECCITY	CHARACTER	30		CIUDAD
16	RECSTATE	CHARACTER	3		ESTADO
17	RECPREFIX	CHARACTER	7		LARGA DISTANCIA
18	RECPHONE	CHARACTER	7		TELEFONO BENEFICIARIO
19	RECZIP	CHARACTER	5		CODIGO POSTAL
20	CURRTYPE	CHARACTER	3		MONEDA DE LA TRANSACCION: 'MEX'

Tabla 1. Registros de datos enviados por Unión del Oeste.

Para los registros trailer (120) véase la tabla 2:

	NOMBRE DEL CAMPO	TIPO	LONG	DEC	DESCRIPCION
1	RECORD_TYPE	CHARACTER	3		IDENTIFICADOR DEL REGISTRO (120)
2	GUIA	CHARACTER	15		NUMERO TOTAL DE REGISTROS SIN CONTAR EL TRAILER.
3	MTCN	CHARACTER	10		NULL
4	TRANSTYPE	CHARACTER	3		NULL
5	ORIGIN	CHARACTER	3		NULL
6	DATETIME	CHARACTER	16		FECHA DE LA TRANSMISION: FORMATO dd-mm-yyyy hh:mm
7	AMOUNTPESOS	NUMERIC	12	2	NULL
8	SENDNAME	CHARACTER	40		MONTO TOTAL DE LAS TRANSACCIONES NUEVAS (110) EN EFECTIVO
9	SENDPREFIX	CHARACTER	7		NUMERO DE REGISTROS NUEVOS (110) EN EFECTIVO
10	SENDPHONE	CHARACTER	7		NUMERO DE REGISTROS NUEVOS (110) EN CHEQUE
11	RECNAME	CHARACTER	40		MONTO TOTAL DE LAS TRANSACCIONES NUEVAS (110) EN CHEQUE
12	REC2NAME	CHARACTER	40		NULL
13	RECSTREET	CHARACTER	40		NULL
14	RECCOLONY	CHARACTER	30		NULL
15	RECCITY	CHARACTER	30		NULL
16	RECSTATE	CHARACTER	3		NULL
17	RECPREFIX	CHARACTER	7		NULL
18	RECPHONE	CHARACTER	7		NULL
19	RECZIP	CHARACTER	5		NULL
20	CURRTYPE	CHARACTER	3		NULL

Tabla 2. Registros tipo trailer enviados por Unión del Oeste.

Los archivos que transmitirá Unión del Oeste hacia MultiPaquetería tendrán formato de texto, cada campo estará entrecorinado y delimitado por comas (","), y cada uno de los registros incluyendo el trailer tendrán una longitud de 400 caracteres.

La información contenida en el campo RECSTATE será obtenida sobre la base de las claves contenidas en la tabla 3:

	<b>CLAVE</b>	<b>DESCRIPCION</b>
1	AGS	Aguascalientes
2	BCN	Baja California Norte
3	BCS	Baja California Sur
4	CAM	Campeche
5	COA	Coahuila
6	COL	Colima
7	CHS	Chiapas
8	CHI	Chihuahua
9	DF	Distrito Federal
10	DGO	Durango
11	GTO	Guanajuato
12	GRO	Guerrero
13	HGO	Hidalgo
14	JAL	Jalisco
15	MEX	Estado de México
16	MIC	Michoacán
17	MOR	Morelos
18	NAY	Nayarit
19	NVL	Nuevo Loen
20	OAX	Oaxaca
21	PUE	Puebla
22	QRO	Querétaro
23	SLP	San Luis Potosí
24	SIN	Sinaloa
25	SON	Sonora
26	TAB	Tabasco
27	TAM	Tamaulipas
28	TLX	Tlaxcala
29	VER	Veracruz
30	YUC	Yucatán
31	ZAC	Zacatecas
32	QOO	Quintana Roo

Tabla 3. Catálogo de Estados.

### 2.3.2 INFORMACIÓN DE MULTIPAQUETERIA HACIA UNION DEL OESTE.

Unión del Oeste necesita llevar un control de las transacciones que son contratadas en sus puntos de venta para diversos fines tales como, dar información a sus clientes acerca de la situación de los servicios que les contratan, cerrar los ciclos de vida de sus transacciones, determinar los importes que se pagarán a MultiPaquetería por la entrega de los envíos, etc. Es por esto que requiere que MultiPaquetería

retroalimente la información de las transacciones, indicando cuáles son los avances que presentan los envíos hasta el momento que se entreguen o se cancelen.

Para garantizar la veracidad de la información que se transmitirá a Unión del Oeste, MultiPaquetería utilizará los datos contenidos en su base de datos centralizada, ya que esto conlleva a disponer de datos actualizados en línea por las áreas operativas que generan los cambios en las situaciones de cada envío. Obviamente que al tener características especiales la información, necesita de cierto tratamiento antes de ser transmitida, este tratamiento se hará en la base de datos instalada en el servidor de este proyecto.

Los status que MultiPaquetería puede mandar a Unión del Oeste serán:

100.- Para informar que el dinero (envío) se ha entregado al beneficiario de la transacción.

101.- Este tipo de registro es mandado como acuse de recibo indicando que MultiPaquetería ha recibido la transacción en cuestión y está procesándola.

102.- Es para informar a Unión del Oeste que se canceló la transacción en cuestión como respuesta a una solicitud por su parte (siempre precedido por un registro 111 enviado por Unión del Oeste).

103.- Para notificar a Unión del Oeste que la dirección proporcionada es equivocada, por lo que no se pudo realizar la entrega del envío.

104.- Para informar a Unión del Oeste que se ha hecho un intento de entregar el dinero, pero no fue posible realizar la entrega por causas como que no estuvo el beneficiario, no aceptó el envío, etc...

105.- Para informar a Unión del Oeste que no se pudo entregar el envío en el segundo intento, debido a la ausencia del destinatario primario o secundario o alguna otra causa. El envío va a la sucursal más cercana para esperar los 10 días más a que lo recoja allí el destinatario.

106.- Se calcula que entre la fecha en que llega una transacción nueva, se intenta entregar el envío un par de veces y los diez días que el envío estará en las instalaciones de MultiPaquetería se llevará un lapso de 16 días, por lo que después de ese lapso si el envío no ha sido entregado o cancelado, automáticamente se debe cancelar en los sistemas de MultiPaquetería y notificar dicha cancelación a Unión del Oeste a través de este tipo de registro.

120 – El registro 120 es clasificado como el ‘trailer’ del archivo. Este registro sirve para validar los datos que se enviarán a Unión del Oeste.

### 2.3.2.1 Frecuencia de transmisión

Para los fines y requerimientos de la información que se transmitirá hacia Unión del Oeste, se ha determinado que solamente será necesario realizar una transmisión al día, la cual será realizada una vez que terminan la mayoría de procesos operativos en MultiPaquetería que podrían afectar los status de los envíos, por lo que la hora asignada será a las 21:00 hrs.

### 2.3.2.2 Formato de la Información.

La nomenclatura que llevarán los archivos a transmitir por parte del MultiPaquetería será bajo las siguientes características:

Supongamos que la fecha de la transmisión es el día 20 de Julio del 2001, el archivo a transmitir será:

21:00 hrs. → mu072001.txt

En donde:

**mu** .- Identifica que es un archivo transmitido de multiPaquetería hacia unión del Oeste.

**0720** .- Es el mes y el día de la transmisión (mmdd).

01.- Se refieren al número de transmisión realizada al día, que como se menciona, será una solamente al día.

Los archivos que transmitirá MultiPaquetería a Unión del Oeste se basarán en la siguiente descripción o layout:

Para los registros de datos véase la tabla 4:

	NOMBRE CAMPO	TIPO	LONG	DEC	DESCRIPCION
1	CAMPO CLAVE	CHARACTER	3		IDENTIFICADOR DEL REGISTRO
2	NUMERO DE GUIA	CHARACTER	12		NUMERO DE TRANSACCION DE MULTIPAQUETERIA
3	FACTURA_DE_UNION	CHARACTER	10		NUMERO DE TRANSACCION DE WESTERN UNION
4	TIPO CHEQUE	CHARACTER	3		'CHQ' = CHEQUE, 'EFE' = EFECTIVO
5	FECHA_DOCUMENTACION	CHARACTER	16		FECHA Y HORA DE LA TRANSACCION, EN FORMATO dd-mm-yyyy hh:mm
6	IMPORTE_TOTAL	NUMERIC	12	2	IMPORTE DEL CHEQUE
7	TIPO_DE_MONEDA	CHARACTER	3		DESCRIBE CUAL TIPO DE MONEDA SE ENTREGA. 1. PESOS = MEX 2. DOLARES = USA
8	FECHA1	CHARACTER	16		FECHA Y HORA DE LA PRIMERA VEZ QUE NO SE PUDO ENTREGAR EL ENVIO: dd-mm-yyyy hh:mm
9	FECHA2	CHARACTER	16		FECHA Y HORA DE LA SEGUNDA VEZ QUE NO SE PUDO ENTREGAR EL ENVIO: dd-mm-yyyy hh:mm
10	FECHA_DE_RECIBO	CHARACTER	16		FECHA Y HORA DE CUANDO SE ENTREGO EL ENVIO dd-mm-yyyy hh:mm
11	CON_NOMBRE	CHARACTER	40		NOMBRE DEL CONSIGNATARIO
12	CON_NOMBRE2	CHARACTER	40		NOMBRE DEL CONSIGNATARIO ALTERNATIVO
13	CON CALLE NUMERO	CHARACTER	40		DIRECCION CALLE Y NUMERO
14	CON COLONIA	CHARACTER	30		COLONIA
15	CON CIUDAD	CHARACTER	30		CIUDAD
16	CON ESTADO	CHARACTER	3		ESTADO
17	CON LARGA DISTANCIA	CHARACTER	7		CLAVE LARGA DISTANCIA
18	CON TELEFONO	CHARACTER	7		NUMERO TELEFONICO
19	CON CODIGO POSTAL	CHARACTER	5		CODIGO POSTAL
20	REM_NOMBRE	CHARACTER	40		NOMBRE DEL REMITENTE
21	NUMERO_DE_ID	CHARACTER	20		NUMERO DE ID DEL CONSIGNATARIO
22	TIPO_DE_ID	CHARACTER	30		TIPO DE ID DEL CONSIGNATARIO
23	QUIEN_RECIBIO	CHARACTER	1		¿CUÁL CONSIGNATARIO LO RECIBIO? 1 = PRIMARIO; 2 = SECUNDARIO

Tabla 4. Registros de datos enviados por MultiPaquetería.

## Reglas:

- Cuando el registro es de tipo 104, siempre debe de haber un valor en el campo 'fecha1'
- Cuando el registro es de tipo 105, siempre debe de haber un valor en el campo 'fecha2' y 'fecha1'
- Cuando el registro es de tipo 100, siempre debe de haber un valor en el campo 'fecha\_de\_recibo'. Si ha habido registros anteriores de 104 y/o 105 de este envío, debe incluir esas fechas en los campos correspondientes (fecha1 y fecha2).
- Cuando el registro es de tipo 106, siempre debe de haber un valor en el campo 'fecha2' y 'fecha1'.
- Los datos de los campos 2,3,4,5,6,7,11,12,13,14,15,16,17,18,19 y 20 vienen del registro correspondiente que se recibió en el archivo de Unión del Oeste.
- El archivo mantendrá una ordenación de los registros por "campo\_clave" primero, y después por 'Tipo\_cheque'.

### 2.3.3 MECANISMOS DE SEGURIDAD DE LA INFORMACIÓN.

Como se ha hecho notar en la primera parte de este capítulo, se utilizará una línea de comunicación privada entre Unión del Oeste y MultiPaquetería, aun así ha surgido la necesidad de que la información transmitida por parte de Unión del Oeste guarde un esquema de seguridad, debido a que en los archivos a transmitir hay información

que involucra dinero y si esa información fuese manipulada antes de ser procesada, podría propiciar problemas muy serios. En cuanto a la información transmitida por parte de MultiPaquetería, no es justificable adoptar medidas mayores de seguridad, ya que lo que contiene es tan solo información sobre las situaciones de los envíos que se están tratando de entregar por su parte, de hecho esta información está disponible en la página que tiene en internet MultiPaquetería.

Por lo anterior, se decide que Unión del Oeste desarrollará un proceso de encriptación el cual será aplicado a los archivos antes de ser transmitidos y otro de desencriptación el cual proporcionará a MultiPaquetería, para aplicar antes de las transmisiones y cuando MultiPaquetería necesite desencriptar los datos de cada transmisión y así poder procesarlos adecuadamente.

**CAPITULO 3. ANALISIS DE PROCESOS A REALIZAR POR PARTE  
DE MULTIPAQUETERIA.**

### **3.1 TRANSMISION DE INFORMACION HACIA REGIONES DE MULTIPAQUETERIA.**

#### **3.1.1 COBERTURA ASIGNADA AL PROYECTO.**

La palabra “cobertura” forma parte del lenguaje diario de cualquier empresa enfocada a aspectos logísticos. Para MultiPaquetería, La cobertura se refiere a todos aquellos lugares en los que se pueden contratar sus servicios como origen y también a los lugares a los que se pueden dirigir los servicios como destino. La cobertura establecida para MultiPaquetería está implementada de acuerdo a estudios logísticos aunados a la capacidad operativa y la infraestructura con la que cuenta, buscando otorgar un servicio con la mayor eficiencia posible a sus clientes.

Para concretar el presente proyecto es necesario considerar la mencionada cobertura y asegurarse de que los servicios contratados en E.U.A. tengan como destino localidades geográficas a las que pueda llegar el servicio de entrega a domicilio de MultiPaquetería, por lo que se proporciona a Unión del Oeste un catálogo de la cobertura manejada por MultiPaquetería. La información se proporciona con la finalidad de que sea implementada en los sistemas que operan en los puntos de venta de Unión del Oeste en E.U.A., y de esta manera asegurar la validez de las transacciones desde el momento de su generación, es decir, todas las

transacciones transmitidas a MultiPaquetería tendrán como destino un punto geográfico contemplado dentro de su capacidad operativa.

El catálogo proporcionado a Unión del Oeste contiene los siguientes datos:

- Código postal.
- Nombre de la población.
- Estado.
- Municipio
- Región (de acuerdo a la definición de MultiPaquetería).
- Cabecera (de acuerdo a la definición de MultiPaquetería).
- Clave iata de la población (de acuerdo a la definición de MultiPaquetería).

De acuerdo a como se organizan las rutas de entrega de MultiPaquetería, el dato más relevante de este catálogo es el código postal, y de hecho será precisamente este dato el punto de validación para saber si el servicio contratado se destina a un punto válido para su entrega.

Dadas estos planteamientos queda en manos de Unión del Oeste el asegurar que las direcciones a las que se destinen los servicios contratados en sus puntos de venta, sean direcciones válidas para MultiPaquetería y de esta manera lograr entregas eficientes y oportunas, conservando así la tendencia de otorgar servicios orientados a la satisfacción total del cliente.

### 3.1.2 DETERMINACION DE REGION DESTINO.

Como se mencionó en su oportunidad, MultiPaquetería está organizado en 22 regiones que cubren la república mexicana, estas regiones fueron definidas de acuerdo a análisis logísticos que permiten establecer puntos de conexión para mover los envíos de una manera óptima. Estas regiones están identificadas por claves iata<sup>4</sup> sobre la base de la tabla 5:

---

<sup>4</sup> Una clave iata es un identificador asignado a una población geográfica específica. Las claves iata están conformadas por tres caracteres alfabéticos.

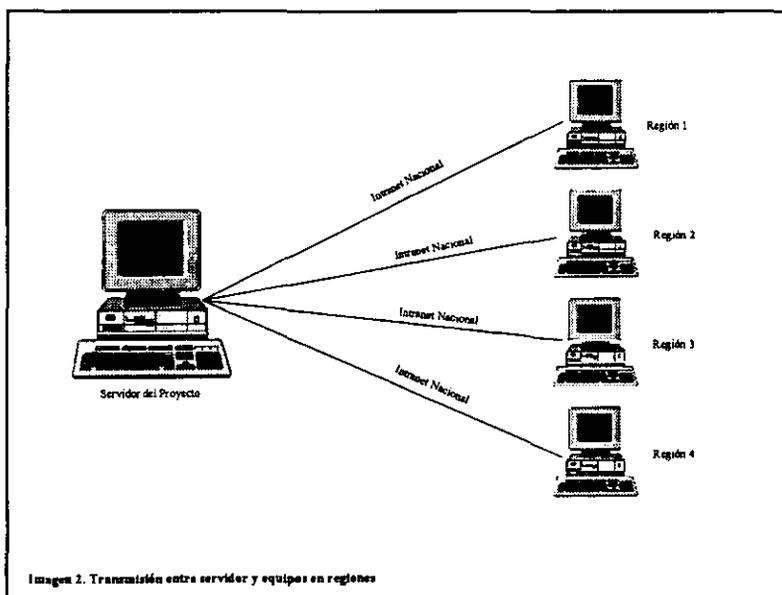
REGION (IATA)	DESCRIPCION
AGU	AGUASCALIENTES
BAJ	BAJIO
CHS	CHIAPAS
CTO	CENTRO
CUW	CHIHUAHUA
CVV	CUERNAVACA
GFO	GOLFO
HMS	HERMOSILLO
IST	ISTMO
MEX	MEXICO
MLA	MORELIA
NTE	NORTE
NTH	MONTERREY
OCC	OCCIDENTE
OWX	OAXACA
PEN	PENINSULAR
QRT	QUERETARO
SIN	CULIACAN
SLP	SAN LUIS POTOSI
TBO	TABASCO
TLC	TOLUCA
TRN	TORREON

Tabla 5. Catálogo de claves lata por región.

Para este proyecto se ha decidido que cada una de estas regiones controle, y administre los envíos que se deban entregar dentro de las poblaciones de su jurisdicción, buscando agilidad en el proceso de entrega, ya que de esta manera cada envío por entregar será generado en el punto geográfico más cercano a su lugar de entrega, evitando de esta manera un traslado de los envíos por puntos innecesarios.

Dado este planteamiento, es necesario que en el servidor del proyecto, se genere un proceso que identifique la región a la que corresponde la entrega de los envíos tomando como base el código postal de la dirección de los beneficiarios en cada uno de los servicios transmitidos por Unión del Oeste. La determinación de la región destino se hace mediante una comparación del código postal del registro contra el catálogo de la cobertura.

Una vez identificadas las regiones destino de cada registro, se deben generar archivos para cada una de las regiones, estos archivos llevan exactamente las mismas características del archivo que originalmente transmite Unión del Oeste y cada uno de esos archivos se debe transmitir automáticamente y de manera encriptada a un equipo operativo designado en cada región para este proyecto, para lo cual se necesita plantear un esquema de transmisión como el mostrado en la imagen 2.



### 3.1.3 FTP COMO MEDIO DE TRANSMISION.

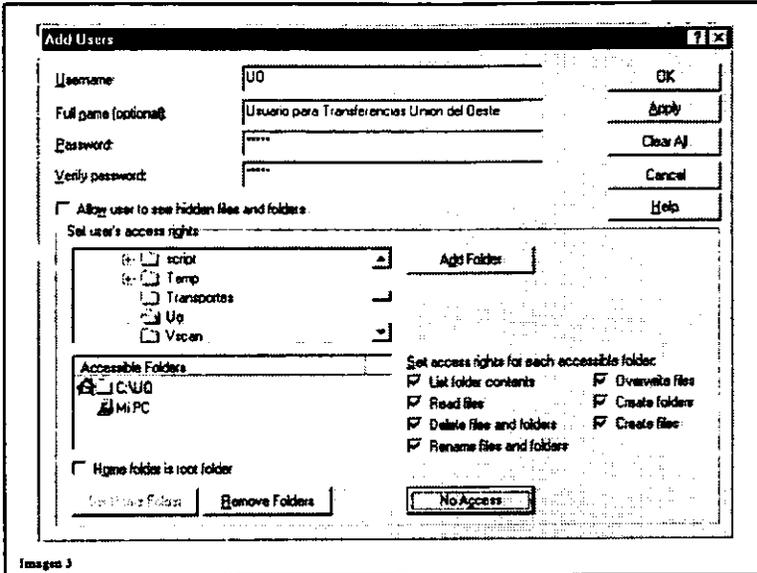
Se ha definido que se utilizará FTP como herramienta para transmitir la información que corresponde a cada una de las regiones, por lo que en cada una de las

máquinas de las regiones se instalará y configurará el mismo software que se utiliza en el servidor del proyecto para hacer esta transmisión hacia equipos con Windows 95.

En la máquina de cada región se configurará un usuario para FTP llamado UO con su respectiva contraseña y un directorio UO en la unidad C:\ que será el predeterminado para el usuario y tendrá todos los privilegios sobre ese directorio. Dentro de este directorio habrá un subdirectorio llamado como la clave iata de la región en cuestión y será ahí donde se vayan depositando los archivos que correspondan. La pantalla para configurar el mencionado usuario es como la mostrada en la imagen 3.

Se debe implementar en el proceso que identifica los registros de cada región, la generación de archivos ftp que se ejecuten de manera automática para transmitir la información generada, estos archivos básicamente contienen datos como el equipo destino, el modo de transmisión y los comandos que transfieran los archivos a procesar. En su oportunidad se mostrará el proceso para generarlos y la muestra de alguno de esos archivos.

Al utilizar la Intranet con la que cuenta MultiPaquetería, la comunicación entre el servidor del Proyecto y cada una de las máquinas de las regiones es transparente, disponiendo de un enlace permanente y seguro entre todos estos equipos, por lo que este esquema de comunicación queda como se ilustró en la imagen 2.



Images 3

## **3.2 INTEGRACION DE ENVIOS A LA BASE DE DATOS CENTRALIZADA DE MULTIPAQUETERIA.**

### **3.2.1 CARGA MASIVA.**

Existen clientes que contratan a MultiPaquetería para transportar un gran volumen de envíos. Estos clientes son catalogados por MultiPaquetería como clientes especiales, a los cuales se les proporcionan beneficios adicionales importantes como por ejemplo, tarifas unitarias independientemente del origen o destino de su paquetería, lo que evita estar calculando una tarifa por cada uno de los paquetes que se vayan a transportar, en realidad la facturación que se les hace a estos clientes dependerá mas bien de que tantos envíos transporten por MultiPaquetería.

A estos clientes se les asigna una clave o identificador único, el cual consta de cuatro dígitos numéricos, y que sirve para el control de la situación del cliente, ya que generalmente a estos clientes se les proporciona crédito y la facturación, cobranza y demás controles se hacen precisamente a través de esta clave.

Dada esta característica y buscando agilidad al momento de recibir nuevos lotes de envíos de estos clientes, se desarrolló un proceso denominado Carga Masiva, cuya finalidad es la de integrar a la base de datos de MultiPaquetería todos estos envíos sin necesidad de realizar grandes capturas manuales.

Para concretar este proceso es necesario que el cliente, proporcione a MultiPaquetería un archivo que contendrá la información relevante de cada envío. Este archivo se rige por algunas prerrogativas como lo son:

Se debe generar desde el control del cliente, un identificador denominado Guía Operativa, la cual es única para cada uno de los envíos. Este identificador es muy importante ya que es a través de él como viajará el envío dentro de MultiPaquetería y se podrá otorgar información, rastreo y demás datos sobre cada uno de los envíos. Este identificador estará formado por una cadena de doce caracteres (en algunos casos pueden ser alfanuméricos) y se formará por la clave asignada al cliente y un número consecutivo de ocho caracteres. Por ejemplo, si se asigna a un cliente la clave 9999, las guías operativas que generará este cliente serán:

9999000000001, 9999000000002, 9999000000003, 9999000000004, etc.

El archivo debe tener un formato de texto, campos de ancho fijo, un registro de encabezado y un registro de resumen bajo las prerrogativas mostradas en las tablas 6, 7 y 8:

No. Campo	Nombre	Tipo	Longitud c/punto decimal	Decimales	Obligatorio	Posición del	Posición al
1	Identificador	Alfanumerico	1		Si	1	1
2	Consecutivo	Númérico	6	0	Si	2	7
3	NumeroCliente	Númérico	14	0	Si	8	21
4	NumeroSucursal	Númérico	4	0	Si	22	25

Tabla 6. Registro con Encabezado del archivo para carga masiva.

No. Campo	Nombre	Tipo	Longitud c/punto decimal	Decimales	Obligatorio	Posición del	Posición al
1	Identificador	Alfanumerico	1		Si	1	1
2	Consecutivo	Númérico	6	0	Si	2	7
3	NombreConsignatario	Alfanumérico	40		Si	8	47
4	CPConsignatario	Númérico	5	0	Si	48	52
5	RFCConsignatario	Alfanumérico	13		Si	53	65
6	CalleNumConsignatario	Alfanumérico	35		Si	66	100
7	ColoniaConsignatario	Alfanumérico	30		Si	101	130
8	TelefonoConsignatario	Alfanumérico	15		No	131	145
9	PoblacionConsignatario	Alfanumérico	40		No	146	185
10	lataConsignatario	Alfanumérico	3		No	186	188
11	ObservacionesGuia	Alfanumérico	50		No	189	238
12	ValorDeclarado	Númérico	8	2	Si	239	246
13	PedidoCliente	Alfanumérico	15		Si	247	261
14	ReferenciaCliente	Alfanumérico	20		No	262	281
15	LongitudPaquete	Númérico	6	3	Si	282	287
16	AnchoPaquete	Númérico	6	3	Si	288	293
17	AltoPaquete	Númérico	6	3	Si	294	299
18	PesoPaquete	Númérico	7	3	Si	300	306
19	TipoServicio	Númérico	1	0	Si	307	307

Tabla 7. Registros con Detalle del archivo para carga masiva.

No. Campo	Nombre	Tipo	Longitud c/punto decimal	Decimales	Obligatorio	Posición del	Posición al
1	Identificador	Alfanumerico	1		Si	1	1
2	Consecutivo	Númérico	6	0	Si	2	7
3	TotalRegistros	Númérico	6	0	Si	8	13

Tabla 8. Registro con Totales del archivo para carga masiva.

A través de una opción del sistema de operación de MultiPaquetería, se integran estos datos en la base de datos central, se inicia el proceso de entrega del envío y se van reflejando todos los acontecimientos relacionadas al mismo.

En este proyecto se decide utilizar este mecanismo de captura para los nuevos envíos de Unión del Oeste, ya que presenta características como las de los clientes especiales y se obtiene el beneficio de agilizar la incorporación de los datos en la base de datos centralizada de MultiPaquetería, por lo que dentro de los procesos que se ejecutarán en el servidor del proyecto, es necesario generar un archivo con las

características necesarias para la Carga Masiva, y transmitirlo al equipo que corresponde a la región 'MEX', ya que esta es la región que se encargará de integrar todos los nuevos envíos para que posteriormente cada región controle los que le correspondan. Como se podrá apreciar, en esta parte del proceso no hay una identificación de la región que procesará cada envío, aquí se hará una sola carga de los nuevos envíos en México, y al estar dentro de la base de datos centralizada, se tendrán disponibles para cualquier parte, lo que permitirá que cada región tenga el rastro inicial de cada nuevo envío que le corresponda procesar.

### 3.2.2 IMPRESION DE CHEQUES Y GENERACION DE PAQUETES.

Unión del Oeste proporciona para este proyecto, un software que permite la impresión de los cheques correspondientes a los envíos que se entregarán. Es importante mencionar que los fondos para el proyecto se obtendrán de una cuenta bancaria a nombre de MultiPaquetería, pero que será fondeada<sup>5</sup> por Unión del Oeste de acuerdo a los requerimientos que generen los nuevos servicios contratados. Esto implica que aun las operaciones que se entregarán en efectivo, generarán un cheque, pero la diferencia estriba en que los cheques generados de las transacciones en efectivo, saldrán impresos a favor de la persona responsable del proyecto en cada región y los cheques generados de transacciones en cheque, saldrán impresos a favor del beneficiario final del servicio. En los servicios contratados en efectivo, el responsable del proyecto en cada región será quién cobre

los cheques y proporcionará el dinero en efectivo a las personas que se destinen a armar el paquete que se entregará finalmente al beneficiario. Estas personas introducirán los cheques o el monto en efectivo en unas mangas diseñadas especialmente para este proyecto y que cumplen con ciertas normas de seguridad, anexarán también un comprobante que estipula el monto exacto del cheque o efectivo que se entreguen y finalmente introducirá esta manga en un sobre para disimular el contenido del paquete.

El software de impresión de cheques es una aplicación que trabaja con una pequeña base de datos Access que trabaja de manera local y que básicamente cumple con la función de imprimir cheques en un formato predefinido por el banco con el que se tiene contratada la cuenta. Esta impresión se realiza con base a los datos encriptados que están contenidos en los archivos que Unión del Oeste transmite a MultiPaquetería. Recordemos que con el proceso que separa los envíos correspondientes a cada región se generan archivos individuales con el mismo formato y encriptación, lo que permite que cada región imprima solamente los cheques que le corresponden.

La aplicación de impresión de cheques tiene una opción para importar los datos del archivo que contiene la información de los cheques a imprimir, esta opción integra los datos a tablas de Access, posteriormente se procede a verificar que el consecutivo que tiene configurado el sistema, concuerde con el número del cheque que se va a

---

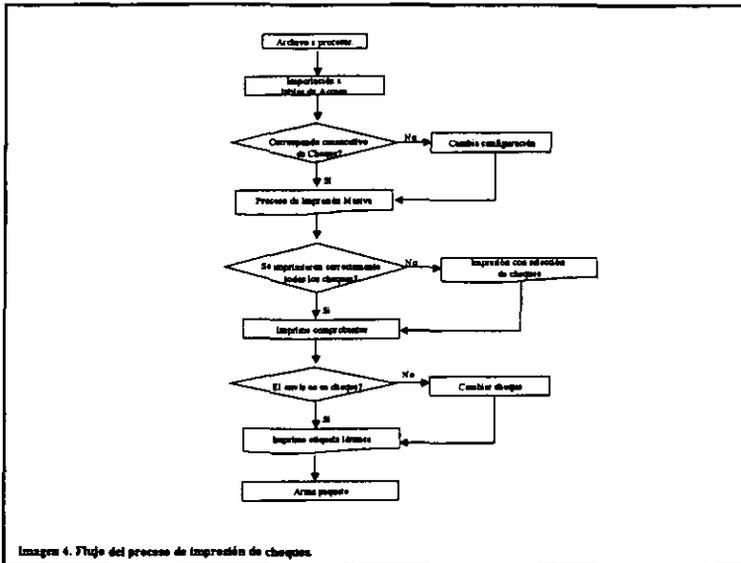
<sup>5</sup> Unión del Oeste hará los depósitos que sean necesarios para que Multipaquetería disponga del dinero necesario para generar los paquetes a entregar.

imprimir, de no corresponderse, se hará la configuración a través de una opción con contraseña que tiene el software. Una vez que esté lista esta parte, se procede a la opción de imprimir masivamente todos los cheques del archivo que se está procesando. En caso de problemas en esta impresión masiva, se puede recurrir a una impresión seleccionando solo aquellos cheques que tuvieron algún problema. Una vez impresos los cheques, se imprimirá desde otra opción del software, el comprobante que se adjuntará en los paquetes, este comprobante contiene datos como el remitente del servicio, el beneficiario y sobre todo el monto a entregar.

Posteriormente se generará desde el sistema de operación de MultiPaquetería, una etiqueta térmica que contiene los datos del envío y un código de barras que contiene la Guía Operativa. Con este código de barras se agiliza mucho el movimiento de los paquetes ya que en muchos procesos por los que pasa el paquete se utilizan scanners que identifican cada uno de los paquetes a través de este código. El hecho de imprimir la etiqueta térmica, también implica que para la base de datos centralizada estos envíos ya se encuentran físicamente en instalaciones de MultiPaquetería y es la pauta para iniciar con su proceso de entrega el cual ya es el que utiliza rutinariamente MultiPaquetería para cualquier envío. En este momento ya se tienen todos los elementos para armar el paquete.

Como se podrá ver, para este momento ya se ha logrado integrar los envíos de este proyecto a la operación normal de MultiPaquetería.

El flujo del proceso de impresión de cheques y armado de paquetes se muestra en la imagen 4.



### 3.3 PROCESOS DE CANCELACIONES.

En el proyecto que estamos desarrollando se contempla la posibilidad de cancelar la entrega de los envíos una vez que se han transmitido por parte del Unión del Oeste. Los motivos que generan la necesidad de cancelar un envío pueden ser por problemas administrativos por parte de Unión del Oeste o por dificultades para hacer la entrega al beneficiario del servicio por parte de MultiPaquetería. Por lo tanto se pueden derivar dos tipos de cancelaciones, las cuales se describen a continuación:

### 3.3.1 CANCELACIONES POR VIGENCIA.

Como política en el desarrollo del proyecto, quedó asentado que MultiPaquetería intentará hacer la entrega del envío dos veces, si al segundo intento no se concreta la entrega, se notifica en el domicilio del beneficiario que el envío quedará bajo resguardo en alguna sucursal de MultiPaquetería para que lo recoja en un lapso no mayor de 10 días. Pasado este lapso y si no se recoge el envío, se prosigue con la cancelación del envío por vigencia, se desmantela el paquete y se cancela el cheque o se deposita el importe en efectivo a la cuenta del proyecto. Para que este proceso se realice de una manera sencilla, se ha determinado que desde el momento en que se transmite el envío, se intenta entregar en un par de ocasiones, se resguarda en la sucursal y se dan los diez días de plazo mencionados, transcurrirá un tiempo de dieciséis días, por lo que esta será la pauta para determinar si el envío se cancela por vigencia.

Cuando se determina que un envío cae en la condición de cancelación por vigencia, se debe de notificar la situación a Unión del Oeste por la noche con el status "106".

En su oportunidad se detallarán los procesos de cancelaciones por vigencia.

### 3.3.2 CANCELACIONES POR SOLICITUD DE UNION DEL OESTE.

Pueden suscitarse varias causas que motiven esta solicitud por parte de Unión del Oeste, por ejemplo: Quizás después de transmitir la contratación del servicio a

MultiPaquetería el cliente decide cancelarlo, se detectan anomalías en la contratación del servicio, o simplemente problemas administrativos.

Independientemente de la causa de cancelación, el mecanismo para solicitarla es el mismo y se detalla a continuación:

Unión del Oeste transmitirá a MultiPaquetería en los archivos cotidianos los registros que requiere se cancelen, la pauta para identificar la solicitud es porque el identificador de estos registros será un "111", a diferencia de los envíos nuevos que traen un "110", MultiPaquetería identificará estos envíos y los almacenará en una tabla dedicada a las cancelaciones.

Ahora bien, el hecho de que se reciba, identifique y almacene un registro que se está solicitando cancelar no implica la seguridad de que así sea, debido a que en realidad depende de la situación que tenga el envío dentro de la operación de MultiPaquetería, ya que por razones obvias, un envío que ya fue entregado no puede ser cancelado y se debe denegar la solicitud. Si el envío está en proceso de entrega, es decir, que ya está en una camioneta para ser entregado, la cancelación también debe ser denegada ya que hasta que regrese la camioneta sabremos si se logró la entrega. Cualquier otra situación del envío asegura que se pueda realizar la cancelación y se notifique a Unión de Oeste en esa misma noche con el status "102".

En su oportunidad se detallarán los procesos de cancelaciones por solicitud de Unión del Oeste.

### **3.4 ACTUALIZACION DE SITUACIONES DE ENVIOS EN MULTIPAQUETERIA.**

El desarrollo de este tema a detalle implicaría un análisis de muchos procesos operativos de MultiPaquetería. Recordemos que uno de los objetivos del presente trabajo es utilizar en la mayor medida posible todos los recursos y procesos utilizados actualmente por MultiPaquetería, por lo que basta con mencionar que MultiPaquetería cuenta con un sistema utilizado en el ámbito nacional que manipula los datos almacenados en su base de datos centralizada, y será a través de este sistema que se registrará cada movimiento que tengan los envíos hasta el momento de su entrega o su cancelación. Es decir, una vez integrados los nuevos envíos de este proyecto a la base de datos de MultiPaquetería, a estos se les dará el mismo tratamiento que a cualquier otro envío que tiene como meta ser entregado.

Esto nos lleva a la conclusión de que dentro de la base de datos de MultiPaquetería se llevará el control de los envíos y será esta la herramienta que nos permita obtener la información para notificar a Unión del Oeste cual es la situación de cada uno de los envíos.

### **3.5 NOTIFICAR STATUS DE ENVIOS A UNION DEL OESTE.**

#### **3.5.1 OBTENCION E INTERPRETACION DE INFORMACION.**

Como se mencionó en el tema anterior, los diferentes movimientos que vayan teniendo los envíos de este proyecto hasta lograr su entrega, son reflejados en la

base de datos centralizada de MultiPaquetería, por lo que de ella serán extraídos los datos necesarios de acuerdo a las características de la información que se describieron en el capítulo correspondiente.

Un punto importante a considerar es que la información requerida por Unión del Oeste no es totalmente compatible con la información generada en la base de datos de MultiPaquetería, por lo que es necesario transformar la información obtenida. Esto se logra utilizando procesos que interpreten los datos obtenidos y los representen de acuerdo a las especificaciones, para lo cual se usa la base de datos contenida en el servidor del proyecto como intermediario para generar la información correctamente. Al hablar de que tendrán que relacionarse estas dos bases de datos, resulta imprescindible establecer un vínculo entre ellas para que se comuniquen de manera eficiente. El establecimiento de este vínculo se detallará en su oportunidad.

Para comprender mejor el punto de la interpretación de información será necesario explicar un concepto muy difundido en MultiPaquetería: Las causas de no entrega. Y esto debido a que a través de su manejo se podrán interpretar los datos contenidos en la base de datos y por consiguiente generar información veraz para Unión del Oeste.

### 3.5.2 CAUSAS DE NO ENTREGA.

Las causas de no entrega<sup>6</sup> pueden ser imputables a MultiPaquetería, al consignatario del envío o a causas ajenas a ambos, como condiciones ambientales, tránsito vehicular, etc.

MultiPaquetería utiliza alrededor de cincuenta causas de no entrega, pero para este proyecto se utilizarán solo algunas de ellas que además nos servirán para generar status para Unión del Oeste que no se tienen contemplados por MultiPaquetería, por ejemplo, el hecho de que se haya intentado entregar uno de los envíos del proyecto pero no se logre la entrega obedece a un status de MultiPaquetería, pero este status podría ser interpretado por tres status diferentes para Unión del Oeste los cuales son mostrados en la tabla 9:

STATUS MULTIPAQUETERIA	STATUS UNION DEL OESTE
No entregado	Primer Intento entrega (104)
No entregado	Segundo Intento entrega(105)
No entregado	Datos Incompletos (103)

Tabla 9. Relación de Status

De primera instancia, resulta difícil identificar el status real del envío de acuerdo al status de MultiPaquetería, por lo que se recurre a una combinación de status + causa de no entrega para interpretar el status que corresponde a Unión del Oeste como se muestra en la tabla 10.

---

<sup>6</sup> MultiPaquetería utiliza este concepto para identificar el motivo por el que no se logró realizar la entrega de un envío.

STATUS MULTIPAQUETERIA	CAUSAS DE NO ENTREGA	STATUS UNION DEL OESTE
No entregado	Aviso de Visita	Primer intento entrega ( 104)
No entregado	Cte. no recibe (envío vencido)	Segundo intento entrega (105)
No entregado	Domicilio equivocado	Datos Incompletos ( 103)

Tabla 10. Combinación de Status y Causas de no Entrega.

Con esta medida queda resuelto el problema, y es a través de comparar status contra causa de no entrega como el proceso identificará el status que deberá transmitir a Unión del Oeste.

Es importante que el personal responsable de introducir los datos en la base de datos tenga bien claro este concepto para que la información sea generada correctamente.

### 3.5.3 GENERACION Y TRANSMISION DE INFORMACION.

Una vez que se cuenta con la información bajo las especificaciones requeridas por Unión del Oeste, se necesita que el mismo proceso que hizo esta interpretación genere el archivo que se transmitirá.

El mecanismo que se utilizará para hacer llegar a Unión del Oeste el archivo requerido será exactamente el mismo que se utiliza para que Unión del Oeste transmita su información a MultiPaquetería, y será también a través de FTP que se logrará concretar la transmisión. Por lo tanto, Unión del Oeste debe realizar la configuración necesaria en su equipo y otorgar a MultiPaquetería un usuario con su

respectiva contraseña para acceder vía FTP y poder depositar los archivos transmitidos.

## **CAPITULO 4. DESARROLLO TECNICO DEL PROYECTO.**

## 4.1 SOFTWARE Y SERVIDOR DEL PROYECTO

Como se mencionó en algún momento, MultiPaquetería ha designado para este proyecto un servidor, el cual será el responsable de las siguientes actividades:

- Tener instalado y configurado el software que lo haga fungir como servidor de FTP.
- Recibir la información que transmite Unión del Oeste.
- Importar de manera automática la información recibida a una base de datos local.
- Generar archivos para cada región, de acuerdo al destino de cada envío.
- Generar un archivo para que la región de México realice una carga masiva hacia la base de datos central.
- Generar los archivos que permitan parametrizar la transmisión de datos a regiones.
- Transmitir automáticamente los archivos a regiones incluyendo el de carga masiva.
- Comunicarse con la base de datos central para obtener las últimas situaciones de cada uno de los envíos del proyecto.
- Activar los procesos que detecten las cancelaciones necesarias y actualicen las situaciones de los envíos en esta situación.
- Integrar e interpretar estas actualizaciones en su base de datos local.
- Generar el archivo que se transmitirá diariamente a Unión del Oeste, informando la situación actualizada de cada uno de sus envíos.

- Transmitir de manera automática el archivo hacia Unión del Oeste diariamente.

Para cumplir con estas funciones, se ha dispuesto que en el equipo designado convivan las siguientes herramientas:

- Windows NT Server como sistema operativo.
- ViewNow como software para Servidor de FTP.
- Oracle 8.1.5.0.0 para NT como manejador de base de datos.
- PL/SQL como lenguaje para programar los procesos mencionados.
- UTL\_FILE como utilidad para el manejo de archivos en formato de texto dentro de PL/SQL.
- Características de NT como el servicio de Scheduler <sup>7</sup>, el manejo de archivos .bat <sup>8</sup>, etc.

No se entrará a detalle a la instalación de cada una de estas herramientas, ya que esto implicaría un desarrollo muy extenso y técnico, el cual desviaría la atención de los objetivos principales del presente trabajo de seminario, por lo que solamente se expondrán los detalles trascendentes que lo ameriten.

---

<sup>7</sup> Servicio que permite la ejecución de programas en horarios programados.

<sup>8</sup> Archivos de procesamiento por lotes que pueden ejecutarse desde el sistema operativo mediante su invocación.

## 4.2 CARACTERISTICAS IMPORTANTES DEL SOFTWARE UTILIZADO.

### 4.2.1 NT.

En NT se necesitará que el servicio del Scheduler se active de manera automática para poder disparar los procesos en los horarios especificados de manera automática. Asegurarnos de esto será en la pantalla de los servicios del panel de control de NT, como se muestra en la imagen 4.

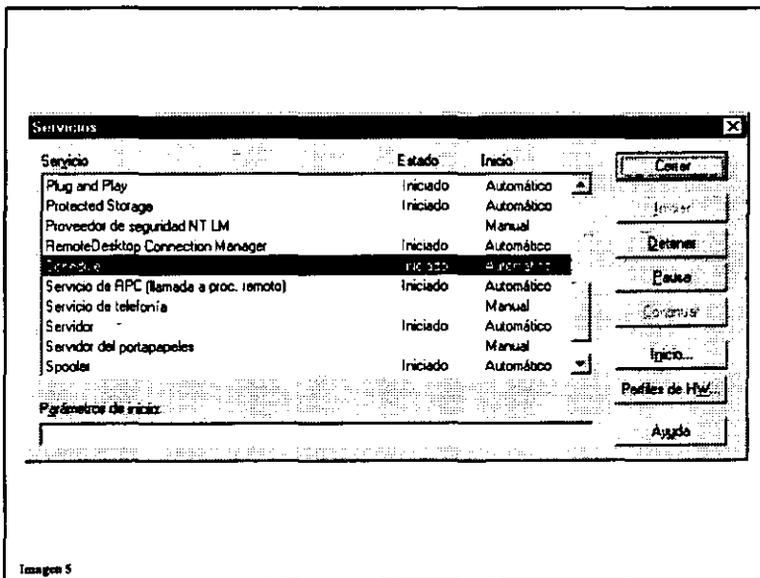


Imagen 5

#### 4.2.2 VIEW NOW.

View Now ofrece una serie de herramientas de comunicación, tales como emuladores, utilerías de transmisión, exportadores de ambientes gráficos, etc.

Para el proyecto solo se instala el Componente denominado FTP SERVER, por lo que al realizar la instalación se hace de manera personalizada. Una vez instalado el software, se procede a crear y configurar un usuario a través de la pantalla que se muestra en el capítulo 3. Este usuario se proporciona a Unión del Oeste para que una vez establecida la comunicación entre su equipo y el de MultiPaquetería, puedan depositar sus archivos en el directorio predeterminado del usuario creado para ellos.

También será necesario asegurar que el servicio que se crea con la instalación esté siempre disponible.

#### 4.2.3 ORACLE 8.1.5.0.0.

En la instalación de Oracle 8.1.5.0.0. para NT, se opta por dejar que el instalador cree una base de datos inicial con los valores predeterminados, lo único que se debe de proporcionar son algunos datos como la instancia<sup>9</sup>.

Se configurará la base de datos para optimizar su desempeño y preparar el esquema de trabajo que se necesita para el proyecto, para esto se realizan actividades como:

---

<sup>9</sup> Nombre de la base de datos Oracle

- Optimizar los tablespaces del sistema.
- Agregar segmentos de rollback<sup>10</sup>.
- Crear tablespaces<sup>11</sup> específicos.
- Crear el usuario que hará todo el manejo de la información.
- Modificar la configuración de la base de datos para el manejo de UTL\_FILE.

Estas actividades se observan y detallan en el Apéndice A referente a la Creación de la base de datos, exceptuando la última, en la cual lo único que se necesita es agregar la siguiente línea:

```
UTL_FILE_DIR=*
```

En el archivo de configuración de la base de datos ubicado en la ruta d:\oracle\admin\unio\pfile y que se llama init.ora. El \* indica que PL/SQL a través de Utl\_File podrá generar y leer archivos en cualquier ubicación del servidor.

Una vez creado el esquema de base de datos que necesitamos es necesario crear las tablas que se utilizan para el manejo de la información y la generación de todos los procesos que ya se han mencionado. Después de un análisis de los requerimientos, se llega a que se necesita crear las siguientes tablas en el usuario unio:

ESTADÍSTICAS DE LA

---

<sup>10</sup> Espacio en disco usado temporalmente para tener la posibilidad de deshacer cambios hechos antes de una actualización definitiva (commit).

- UO\_TRANSF.- Esta tabla se utiliza para incorporar los nuevos envíos transmitidos por Unión del Oeste (110).
- UO\_REGRESO.- En esta tabla se registran los cambios en las situaciones de los envíos, obtenidos desde la base de datos central de MultiPaquetería.
- ARCHIVOS\_PROCESO.- Esta tabla guarda una bitácora de los archivos que se esperan recibir por parte de Unión del Oeste y los que se van procesando.
- UO\_CANCELACIONES.- Esta tabla se utiliza para registrar las solicitudes de cancelación por parte de Unión del Oeste que pueden venir en los archivos transmitidos (111).
- UO\_CANCELACIONES\_PROBLS.- Si se presenta algún problema para incorporar las solicitudes de cancelación, se registra en esta tabla el error generado.
- UO\_CODIGO\_IATA.- En esta tabla se almacena la cobertura asignada al servicio.
- UO\_ERROR.- En esta tabla se registran los errores en caso de que haya algún problema al tratar de acceder a algún archivo transmitido por Unión del Oeste.
- UO\_PROBLS.- En esta tabla se almacenarán los errores presentados en caso de que alguno o algunos de los registros de los archivos transmitidos no puedan ser leídos correctamente.
- UO\_REG\_PROCESA.- En esta tabla se establecerá una relación de la región que procesará los envíos en caso de que alguna de las regiones se imposibilite para el manejo de los envíos del proyecto. Idealmente habrá una relación uno a uno en esta tabla.

---

<sup>11</sup> Espacio predispuesto en disco para crear objetos de la base de datos (tablas, índices, vistas, etc.).

La descripción de cada una de las tablas así como sus llaves primarias<sup>12</sup> puede observarse en el Apéndice B relativo a la Creación de tablas del usuario.

#### 4.2.4 PLSQL y UTL\_FILE.

Con la utilización conjunta de estos dos elementos, se pretende no solamente integrar la información de las nuevas transmisiones de Unión del Oeste y generar la información que se les enviará de regreso, sino también generar los archivos que permitan lograr la correcta transmisión de la información.

Es decir, que con UTL\_FILE se generan archivos con extensión bat y archivos con extensión ftp<sup>13</sup>, los cuales serán ejecutados a través del scheduler de NT, logrando así la automatización en la ejecución de los procesos y en la transmisión de datos.

---

<sup>12</sup> Una llave primaria es una validación para que la tabla tenga datos únicos en el o los campos especificados y agiliza las búsquedas sobre la tabla ya que es a la vez un índice.

<sup>13</sup> Estos son archivos que brindan parámetros para establecer comunicación y transmitir archivos vía FTP.

### **4.3 DEFINICION Y DESARROLLO DE PROCESOS.**

De las tareas que hemos descrito que están a cargo del servidor, es necesario conjuntar e integrar algunas de ellas dentro del mismo proceso, por ejemplo, se detecta que resulta necesario que en el primer proceso a desarrollar se conjunten las siguientes tareas:

- Importar de manera automática la información recibida a una base de datos local.
- Generar archivos para cada región, de acuerdo al destino de cada envío.
- Generar un archivo para que la región de México realice una carga masiva hacia la base de datos central.
- Generar los archivos que permitan parametrizar la transmisión de datos a regiones.

Para esto, se parte de la base de que las comunicaciones y el equipo NT como servidor de FTP trabajan correctamente y se logra satisfactoriamente la transmisión de archivos por parte de Unión del Oeste. Por lo que una vez teniendo los archivos disponibles en el directorio predeterminado del usuario de FTP designado a Unión del Oeste se comienzan las tareas arriba mencionadas.

#### 4.3.1 IMPORTACION Y DISTRIBUCION DE INFORMACION.

En esta fase se utiliza `UTL_FILE`, para poder visualizar uno a uno cada registro contenido en las transmisiones de Unión del Oeste, y de esta manera poder insertar en la tabla correspondiente, sea envío nuevo o solicitud de cancelación. Por lo que antes que nada es necesario implementar en el proceso la declaración de variables y de objetos UTL que utilizaremos, como muestra la tabla 11.

```

DECLARE
  Varchivo          UTL_FILE.FILE_TYPE;
  VarchExis        UTL_FILE.FILE_TYPE;
  VarchCons         UTL_FILE.FILE_TYPE;
  VarchValid        UTL_FILE.FILE_TYPE;

  vTipo_Registro   varchar2(3);
  vNumero_Guia     varchar2(15);
  vNumero_Ident    varchar2(12);
  vTranstype       varchar2(3);
  vOrigin           varchar2(3);
  vDatetime        varchar2(16);
  vAmount_Pesos    varchar2(12);
  vRemitente       varchar2(40);
  vPrefijo_Rte     varchar2(7);
  vNumero_Rte      varchar2(15);
  vNombre_Bnro     varchar2(40);
  vNombre2_Bnro    varchar2(40);
  vDireccion        varchar2(40);
  vColonia          varchar2(30);
  vCiudad           varchar2(30);
  vEstado           varchar2(3);
  vPrefijo          varchar2(7);
  vTelefono         varchar2(15);
  vCodigo_Postal   varchar2(5);
  vTipo_Moneda     varchar2(3);

  vCount_Arch      number(1);
  vNombre_Arch     varchar2(12);
  vBuffer           varchar2(500);
  verror_cod       varchar2(15);
  verror_msg       varchar2(80);
  v_cuenta         number := 0;
  v_guarda         number := 1;
  v_totreg         number(5);
  v_totefe         number(5);
  v_totchq         number(5);
  v_impefe         number(10);
  v_impchq         number(10);
  v_iniproc        number(5);
  v_finproc        number(5);
  v_numarch        varchar2(2);

```

Tabla 11. Definición de variables y manejadores de archivos UTL.

La distinción para identificar el día y la hora de la transmisión, y darle el tratamiento correspondiente de acuerdo a los horarios mencionados se muestra en la tabla 12 y se rige por los siguientes preceptos:

- Si es sábado antes de las 8:00 hrs. se comienza en el archivo 01 y se termina en el 01. En caso contrario se comienza en el 02 y se termina en el 02.
  
- Si no es sábado y es antes de las 15 hrs. se comienza en el archivo 01 y se termina en el 02. Si es mas tarde a las 15 hrs., se comienza en el archivo 03 y se termina en el 03.
  
- Una vez identificados los archivos que se pretenden procesar de acuerdo a este análisis de la fecha de transmisión se insertan sus nombres en la tabla correspondiente. Se pretende procesar estos y todos los archivos pendientes de procesar.

```

if upper(to_char(sysdate,'dy')) = 'SÁB' then
  if to_number(to_char(sysdate,'hh24')) <= 8 then
    v_iniproc := 1;
    v_finproc := 1;
    v_numarch := '01';
  else
    v_iniproc := 2;
    v_finproc := 2;
    v_numarch := '02';
  end if;
else
  if to_number(to_char(sysdate,'hh24')) <= 15 then
    v_iniproc := 1;
    v_finproc := 2;
    v_numarch := '01';
  else
    v_iniproc := 3;
    v_finproc := 3;
    v_numarch := '02';
  end if;
end if;
FOR vCount_Arch in v_iniproc .. v_finproc LOOP
  BEGIN
    Vnombre_Arch := 'UM'||to_char(sysdate,'mmdd')||'0'||to_char(vCount_Arch)||'.TXT';
    insert into archivos_proceso values (vNombre_Arch, null,null);
  EXCEPTION
    When others then null;
  END;
END LOOP;

```

Tabla 12 Distinción de día y hora de la transmisión para identificar archivos a procesar.

En esta parte del proceso se implementa un cursor<sup>14</sup> que obtendrá todos los archivos pendientes de procesar, incluyendo él o los que correspondan a la corrida actual. Este cursor puede ser analizado en la tabla 13.

```

DECLARE
cursor cArchProc is
  select nombre_archivo from archivos_proceso where procesado is null;
BEGIN
  for vArchProc in cArchProc loop
    BEGIN
      v_totreg := 0;
      v_totefe := 0;
      v_totchq := 0;
      v_impefe := 0;
      v_impchq := 0;
      vArchivo := UTL_FILE.FOPEN ('d:\UnionO',vArchProc.nombre_archivo,r');
    LOOP
      BEGIN
        UTL_FILE.GET_LINE(vArchivo,vBuffer);
        VTipo_Registro := SUBSTR(vBuffer, 2, INSTR(vBuffer,;,1,1) - 3);
        Vnumero_Guia := SUBSTR(vBuffer, INSTR(vBuffer,;,1,1) + 2, INSTR(vBuffer,;,1,2) - INSTR(vBuffer,;,1,1) - 3);
        Vnumero_Ident := SUBSTR(vBuffer, INSTR(vBuffer,;,1,2) + 2, INSTR(vBuffer,;,1,3) - INSTR(vBuffer,;,1,2) - 3);
        Vranstypc := SUBSTR(vBuffer, INSTR(vBuffer,;,1,3) + 2, INSTR(vBuffer,;,1,4) - INSTR(vBuffer,;,1,3) - 3);
        Vorigin := SUBSTR(vBuffer, INSTR(vBuffer,;,1,4) + 2, INSTR(vBuffer,;,1,5) - INSTR(vBuffer,;,1,4) - 3);
        Vdatetime := SUBSTR(vBuffer, INSTR(vBuffer,;,1,5) + 2, INSTR(vBuffer,;,1,6) - INSTR(vBuffer,;,1,5) - 3);
        VAmount_Pesos := SUBSTR(vBuffer,INSTR(vBuffer,;,1,6) + 1, INSTR(vBuffer,;,1,7) - INSTR(vBuffer,;,1,6) - 1);
        VRemitente := SUBSTR(vBuffer, INSTR(vBuffer,;,1,7) + 2, INSTR(vBuffer,;,1,8) - INSTR(vBuffer,;,1,7) - 3);
        vPrefijo_Rte := SUBSTR(vBuffer, INSTR(vBuffer,;,1,8) + 2, INSTR(vBuffer,;,1,9) - INSTR(vBuffer,;,1,8) - 3);
        vNumero_Rte := SUBSTR(vBuffer, INSTR(vBuffer,;,1,9) + 2, INSTR(vBuffer,;,1,10) - INSTR(vBuffer,;,1,9) - 3);
        vNombre_Bnro := SUBSTR(vBuffer,INSTR(vBuffer,;,1,10) + 2, INSTR(vBuffer,;,1,11) - INSTR(vBuffer,;,1,10) - 3);
        vNombre2_Bnro := SUBSTR(vBuffer, INSTR(vBuffer,;,1,11) + 2, INSTR(vBuffer,;,1,12) - INSTR(vBuffer,;,1,11) - 3);
        vDireccion := SUBSTR(vBuffer, INSTR(vBuffer,;,1,12) + 2, INSTR(vBuffer,;,1,13) - INSTR(vBuffer,;,1,12) - 3);
        vColonia := SUBSTR(vBuffer, INSTR(vBuffer,;,1,13) + 2, INSTR(vBuffer,;,1,14) - INSTR(vBuffer,;,1,13) - 3);
        vCiudad := SUBSTR(vBuffer, INSTR(vBuffer,;,1,14) + 2, INSTR(vBuffer,;,1,15) - INSTR(vBuffer,;,1,14) - 3);
        vEstado := SUBSTR(vBuffer, INSTR(vBuffer,;,1,15) + 2, INSTR(vBuffer,;,1,16) - INSTR(vBuffer,;,1,15) - 3);
        vPrefijo := SUBSTR(vBuffer, INSTR(vBuffer,;,1,16) + 2, INSTR(vBuffer,;,1,17) - INSTR(vBuffer,;,1,16) - 3);
        vTelefono := SUBSTR(vBuffer, INSTR(vBuffer,;,1,17) + 2, INSTR(vBuffer,;,1,18) - INSTR(vBuffer,;,1,17) - 3);
        vCodigo_Postal := SUBSTR(vBuffer, INSTR(vBuffer,;,1,18) + 2, INSTR(vBuffer,;,1,19) - INSTR(vBuffer,;,1,18) - 3);
        vTipo_Moneda := SUBSTR(vBuffer, INSTR(vBuffer,;,1,19) + 2, 3);
      END;
    END LOOP;
  END LOOP;
END;

```

Tabla 13. Cursor para abrir y leer cada archivo a procesar.

Dentro de este ciclo se declara un nuevo ciclo que lee registro a registro los datos del archivo que se está procesando. Es decir, se genera un ciclo para identificar cada archivo que se vaya a procesar, pero por cada archivo que se va procesando se genera un nuevo ciclo para leer línea a línea el archivo y los datos leídos se van

<sup>14</sup> Un cursor es una consulta que mantiene los datos obtenidos en memoria para poder trabajar con ellos.

almacenando en variables para que a través de estas se haga la inserción de los datos en las tablas correspondientes.

Posteriormente se identifica si el envío es nuevo, o es una solicitud de cancelación y se hace la inserción del registro en la tabla que corresponda como se muestra en la tabla 14. Además se validan los datos del archivo, comparándolo con el último registro del archivo (120), que como ya se ha mencionado, es el trailer con los totales esta validación se aprecia en la tabla 15.

```

if vTipo_Registro in ('110','111') then
  v_totreg := v_totreg + 1;
  if vTipo_Registro = '110' then
    if vTransType = 'EFE' then
      v_totefe := v_totefe + 1;
      v_impefe := v_impefe + to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,',',1,1)-1));
    else
      v_totchq := v_totchq + 1;
      v_impchq := v_impchq + to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,',',1,1)-1));
    end if;
  insert into uo_transf (tipo_registro, numero_guia, numero_ident, transtype,
    origin, datetime, amount_pesos, remitente, prefijo_rte,
    numero_rte, nombre_bvro, nombre2_bvro, direccion, colonia,
    ciudad, estado, prefijo, telefono, codigo_postal,
    tipo_moneda, fecha_trans, archivo_origen
    )
    values (vTipo_Registro, vNumero_Guia, vNumero_Ident, vTranstype, vOrigin,
      to_date(vDatetime, 'dd-mm-yyyy hh24:mi'),
      to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,',',1,1)-1)),
      vRemitente, vPrefijo_Rte, vNumero_Rte, vNombre_Bvro, vNombre2_Bvro,
      vDireccion, vColonia, vCiudad, vEstado, vPrefijo, vTelefono,
      vCodigo_Postal, vTipo_Moneda, sysdate, vNombre_Arch
    );
  elsif vTipo_Registro = '111' then
    insert into uo_cancelaciones (tipo_registro, numero_guia, datetime,
      fecha_trans, archivo_origen
    )
    values (vTipo_Registro, vNumero_Guia, to_date(vDatetime, 'dd-mm-yyyy hh24:mi'), sysdate, vNombre_Arch);
  end if;
elsif vTipo_Registro = '120' then
  if to_number(vNumero_Guia) <> v_totreg or
    to_number(substr(vRemitente,1,instr(vRemitente,',',1,1)-1)) <> v_impefe or
    to_number(vPrefijo_Rte) <> v_totefe or
    to_number(vNumero_Rte) <> v_totchq or
    to_number(substr(vNombre_Bvro,1,instr(vNombre_Bvro,',',1,1)-1)) <> v_impchq then
    delete uo_transf where archivo_origen = vNombre_Arch;
  commit;
  IF NOT UTL_FILE.IS_OPEN(vArchCons) THEN
    VarchCons := UTL_FILE.FOPEN ('c:\uo\Inconsis','IC'||to_char(sysdate,'mmdd')||v_numarch||'.TXT','w');
  END IF;
  vbuffer := vNombre_Arch;
  UTL_FILE.PUT_LINE(vArchCons,vBuffer);
end if;
end if;

```

Tabla 14. Identificación e inserción del tipo de registro.

```

elsif vTipo_Registro = '120' then
  if to_number(vNumero_Guia) <> v_totreg or
  to_number(substr(vRemitente,1,instr(vRemitente,',',1,1)-1)) <> v_impefe or
  to_number(vPrefijo_Rte) <> v_totefe or
  to_number(vNumero_Rte) <> v_totchq or
  to_number(substr(vNombre_Bnro,1,instr(vNombre_Bnro,',',1,1)-1)) <> v_impchq then
    delete uo_transf where archivo_origen = vNombre_Arch;
    commit;
    IF NOT UTL_FILE.IS_OPEN(vArchCons) THEN
      VArchCons := UTL_FILE.FOPEN ('c:\uo\Inconsis',IC||to_char(sysdate,'mmdd')||v_numarch||'.TXT','w');
    END IF;
    Vbuffer := vNombre_Arch;
    UTL_FILE.PUT_LINE(vArchCons,vBuffer);
  end if;
end if;

```

Tabla 15. Validación de datos contra registro trailer.

En esta parte del proceso se hace una validación del número de registros y los importes de cada registro contra el trailer del archivo, en caso de que haya discrepancias, se elimina la información que se había insertado en la tabla y se genera un archivo que guarda los nombres de los archivos que tuvieron inconsistencias en la corrida actual.

Una vez identificado el archivo que se pretende procesar, se intenta abrir a través de UTL\_FILE, pero si se genera una excepción por la inexistencia del archivo, se genera un archivo en el que se escribirán los nombres del o los archivos que se debieron haber procesado y que no estuvieron disponibles. Se pretende hacer llegar estos archivos a Unión del Oeste para notificarles las inconsistencias y cuando no transmitieron archivos, esto se puede apreciar en la tabla 16.

```

Exception
when utl_file.read_error or utl_file.invalid_filehandle or utl_file.invalid_operation or utl_file.internal_error then
  if not utl_file.is_open(vArchExis) then
    vArchExis := UTL_FILE.FOPEN ('c:\uo\Existe','EX'||to_char(sysdate,'mmdd')||v_numarch||'.TXT','w');
  end if;
when others then null;
end;

```

Tabla 16. Generación de log de error al no encontrar archivo a procesar.

Una vez que ya se tiene la nueva información en la base de datos, se procede a identificar a que región corresponde cada nuevo registro, para lo cual se debe obtener una relación de las regiones, la cual se hace a partir de la tabla de coberturas a través del cursor mostrado en la tabla 17.

```
cursor cREG is
select a.region, b.reg_procesa from uo_codigo_ata a, uo_reg_procesa b
where a.region = b.region
group by a.region, b.reg_procesa;
```

Tabla 17. Identificación de región destino.

Una vez identificadas las regiones se procesarán una por una, es decir, por cada región se identificarán los registros que le corresponden de acuerdo a la última carga de información.

Se declaran cursores que obtengan primero, el número de nuevos registros que le corresponden a la región (esto para saber si es necesario crear el archivo que le corresponde a la región o no) y en segundo lugar se obtienen los datos necesarios de estos nuevos registros para escribir registro a registro el archivo de la región que se está procesando como se muestra en la tabla 18.

```

BEGIN
  VarcMasa := UTL_FILE.FOPEN ('c:\uo\Masiva',uo)||to_char(sysdate,'ddrmm')||v_numarch||'.txt' , 'w');
  VbufMasa := 'H000001      73587309      uo||to_char(sysdate,'ddrmmyy)';
  UTL_FILE.PUT_LINE(vArcMasa,vBufMasa);
  VarcMain := UTL_FILE.FOPEN ('c:\uo\bata','MainBat.bat' , 'w');
  VbufMain := 'echo Transfiriendo archivos a Regiones';
  UTL_FILE.PUT_LINE(vArcMain,vBufMain);
  FOR vREG IN cREG LOOP
    BEGIN
      DECLARE
        CURSOR cHayReg IS
          SELECT COUNT(*) FROM uo_TRANSF
          WHERE CODIGO_POSTAL IN (SELECT CODIGO_POSTAL
          FROM uo_CODIGO_IATA
          WHERE REGION = vREG.REGION)
          AND FECHA_TRANS > (sysdate - 0.125);
        CURSOR cSORT IS
          SELECT
            A.TIPO_REGISTRO,
            A.NUMERO_GUIA,
            A.NUMERO_IDENT,
            A.TRANSTYPE,
            A.ORIGIN,
            A.DATETIME,
            A.AMOUNT_PESOS,
            A.REMITENTE,
            A.PREFIJO_RTE,
            A.NUMERO_RTE,
            A.NOMBRE_BNRO,
            A.NOMBRE2_BNRO,
            A.DIRECCION,
            A.COLONIA,
            A.CIUDAD,
            A.ESTADO,
            A.PREFIJO,
            A.TELEFONO,
            A.CODIGO_POSTAL,
            A.TIPO_MONEDA,
            B.CLAVE_IATA
          FROM UO_TRANSF A, UO_CODIGO_IATA B
          WHERE A.CODIGO_POSTAL = B.CODIGO_POSTAL AND
            A.CODIGO_POSTAL IN (SELECT CODIGO_POSTAL
            FROM UO_CODIGO_IATA
            WHERE REGION = vREG.REGION)
          AND FECHA_TRANS > (sysdate - 0.125)
          GROUP BY A.TIPO_REGISTRO,
            A.NUMERO_GUIA,
            A.NUMERO_IDENT,
            A.TRANSTYPE,
            A.ORIGIN,
            A.DATETIME,
            A.AMOUNT_PESOS,
            A.REMITENTE,
            A.PREFIJO_RTE,
            A.NUMERO_RTE,
            A.NOMBRE_BNRO,
            A.NOMBRE2_BNRO,
            A.DIRECCION,
            A.COLONIA,
            A.CIUDAD,
            A.ESTADO,
            A.PREFIJO,
            A.TELEFONO,
            A.CODIGO_POSTAL,
            A.TIPO_MONEDA,
            B.CLAVE_IATA;

```

Tabla 18. Selección de registros por cada región.

Independientemente de la región que se procese, se irán agregando los registros en el archivo que se utilizará para realizar la Carga Masiva hacia la base de datos central de MultiPaquetería por parte de la región de México.

En esta parte del proceso se hace la escritura de cada registro en el archivo de texto de la región que se está procesando a través de un ciclo que barre cada registro que corresponde, al igual que el registro para el archivo de la Carga Masiva. Al finalizar el ciclo, se escribe el registro trailer del archivo, el cual, como ya se comentó con anterioridad, contiene un resumen de los datos que contiene el archivo y se cierra el archivo de la región. Como se observa en el segmento de código de la tabla 19.

```

BEGIN
  Open cHayReg;
  fetch cHayReg into vHayDatos;
  Close chayReg;
  If vHayDatos > 0 then
    VArchReg := UTL_FILE.FOPEN ('c:\uo\regiones\||vREG.REGION,'WM'||to_char(sysdate,'mmdd')||v_numarch||'.TXT','w');
    FOR vSORT IN cSORT LOOP
      BEGIN
        if vSORT.TRANSTYPE = 'EFE' then
          vEfectivos := vEfectivos + 1;
          vImpEfectivos := vImpEfectivos + vSORT.AMOUNT_PESOS;
        else
          vCheques := vCheques + 1;
          vImpCheques := vImpCheques + vSORT.AMOUNT_PESOS;
        end if;
        vRegistros := vRegistros + 1;
        vTipo_Registro1 := vSORT.Tipo_Registro;
        vNumero_Guia1 := vSORT.Numero_Guia;
        vNumero_Ident1 := vSORT.Numero_Ident;
        vTranstype1 := vSORT.Transtype;
        vOrigin1 := vSORT.Origin;
        vDatetime1 := to_char(vSORT.Datetime,'dd-mm-yyyy hh24:mi');
        vAmount_Pesos1 := to_char(vSORT.Amount_Pesos);
        vRemitente1 := vSORT.Remitente;
        vPrefijo_Rtel := vSORT.Prefijo_Rte;
        vNumero_Rtel := vSORT.Numero_Rte;
        vNombre_Bnro1 := vSORT.Nombre_Bnro;
        vNombre2_Bnro1 := vSORT.Nombre2_Bnro;
        vDireccion1 := vSORT.Direccion;
        vColonial := vSORT.Colonia;
        vCiudad1 := vSORT.Ciudad;
        vEstado1 := vSORT.Estado;
        vPrefijo1 := vSORT.Prefijo;
        vTelefono1 := vSORT.Telefono;
        vCodigo_Postal := vSORT.Codigo_Postal;
        vTipo_Monedal := vSORT.Tipo_Moneda;
        vIata1 := vSORT.Clave_fata;
        vBuffer1 := '||vTipo_Registro1||'";

```



scheduler de NT, este se encargará de llamar a cada uno los archivos ftp de las regiones.

```

VarchBat := UTL_FILE.FOPEN ('c:\uo\bats',vREG.REGION||'.ftp',w);
VBufbat := 'open uo_||lower(vREG.REG_PROCESA);
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'uo';
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'uo';
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'prompt off';
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'cd ||lower(vREG.REGION);
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'cd \uo\regiones\||lower(vREG.REGION);
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
VBufbat := 'put wm||to_char(sysdate,'mmdd')||v_numarch||'.txt';
UTL_FILE.PUT_LINE(vArchBat,vBufbat);
IF UTL_FILE.IS_OPEN(vArchBat) THEN
    UTL_FILE.FCLOSE (vArchBat);
END IF;
VbufMain := 'ftp -s:c:\uo\bats\||lower(vREG.REGION)||'.ftp';
UTL_FILE.PUT_LINE(vArchMain,vBufMain);
VbufMain := 'move c:\uo\regiones\||lower(vREG.REGION)||'wm'||to_char(sysdate,'mmdd')||v_numarch||'.txt
c:\uo\respaldos\||lower(vREG.REGION);
UTL_FILE.PUT_LINE(vArchMain,vBufMain);

```

Tabla 20. Generación de archivo para FTP.

Para finalizar el proceso se deben de terminar los archivos que lo requieran para posteriormente cerrarlos y dar por terminado el proceso. En este momento ya se tienen los archivos de cada una de las regiones, el archivo para la Carga Masiva, y los archivos que permitirán realizar la transmisión, por lo que ahora resta precisamente invocar el archivo que llamaremos transac.bat, que se encarga de realizar todo el proceso mencionado.

```

echo Procesando nuevas transacciones de Unión del Oeste
sqlplus -s uniono/uniono @ c:\uo\scripts\uo_regiones
echo Copiando archivos a Regiones
c:
cd \uo\bats
call MainBat

```

Tabla 21. Archivo transac.bat para disparar todo el proceso expuesto.

Por lo tanto, lo que se necesita es ejecutar automáticamente el archivo bat que se muestra en la tabla 21, en donde, primero se ejecuta el script<sup>15</sup> que contiene todo el código expuesto y posteriormente ejecuta el archivo MainBat, que también es un archivo .bat que se genera en el script que se ejecuta previamente. Un ejemplo de cómo quedaría el MainBat.bat se muestra en la tabla 22.

```
echo Transfiriendo archivos a Regiones
ftp -s:c:\uo\bats\mex.ftp
ftp -s:c:\uo\bats\baj.ftp
ftp -s:c:\uo\bats\cto.ftp
echo Transfiriendo archivo para carga masiva a Internacional
ftp -s:c:\uo\bats\masiva.ftp
```

Tabla 22. Ejemplo del archivo MainBat.bat

Recordemos que los archivos ftp fueron creados por el script que se ejecuta con anterioridad. En este caso, se detectó información que corresponde a las regiones: México, Bajío y Centro. Para lograr que esto se ejecute automáticamente se debe utilizar el scheduler del servidor a través del comando at como se muestra en la tabla 23.

```
at 14:00 /interactive /every:1,ma,mi,j,v "c:\uo\scripts\transac.bat"
at 14:30 /interactive /every:s "c:\uo\scripts\transac.bat"
at 6:00 /interactive /every:s "c:\uo\scripts\transac.bat"
at 19:00 /interactive /every:1,ma,mi,j,v "c:\uo\scripts\transac.bat"
```

Tabla 23. Configuración del scheduler de NT

Con esto se logra que se ejecute automáticamente el archivo transac.bat en los horarios que ya se expusieron en su oportunidad. El script completo de PL/Sql se puede apreciar en el Apéndice C relativo al Proceso de incorporación y distribución de Información.

<sup>15</sup> Un script es un archivo que contiene código de PL/SQL, generalmente tiene extensión .sql y se puede ejecutar

### 4.3.2 COMUNICACIÓN CON BASE DE DATOS CENTRAL.

Para establecer esta comunicación se utiliza un concepto implementado por Oracle denominado Data Base Link, cuya función es definir una conexión dinámica a una base de datos que reside en otro equipo. Un DBLink, como le llaman, puede ser de dos maneras: Público y privado.

**DBLink Público.**- Para definir un DBLink público es necesario que se especifique que así va a ser desde su creación a través del parámetro PUBLIC y debe ser creado por el usuario SYSTEM, que es el administrador de la base de datos. Lo que se logra con la creación de un DBLink público es que cualquier usuario de la base de datos disponga de esta línea de comunicación con la base de datos que se especifica. Un ejemplo de creación de un DBLink público se observa en la tabla 24.

```
Create public database link liga connect to usuario identified by passusuario using 'cadena_conexion'
```

Tabla 24. Ejemplo de Creación de DBLink público

**DBLink privado.**- Este tipo de DBLink a diferencia del público, se puede crear con cualquier usuario de la base de datos, pero la limitante que se tiene es que solo el usuario que lo crea puede utilizarlo. Un ejemplo para crear un DBLink privado se muestra en la tabla 25.

```
Create database link liga connect to usuario identified by passusuario using 'cadena_conexion'
```

Tabla 25. Ejemplo de Creación de DBLink privado.

---

desde cualquier editor SQL mediante el comando execute o el carácter @.

Una vez creado el DBLink, bastará con hacer referencia a los objetos del usuario especificado de la otra base de datos utilizando el nombre del objeto adjuntándole el carácter "@" y el nombre del DBLink que se está utilizando. El modo de utilizar un DBLink se muestra en la tabla 26.

```
Select * from tabla1@liga;
```

Tabla 26. Utilización de un DBLink en una consulta.

En donde tabla1 es una tabla cuyo dueño es el usuario que se especificó al momento de crear el DBLink llamado "liga".

### 4.3.3 PROCESO DE CANCELACIONES.

Como se ha venido comentando a lo largo del presente trabajo, uno de los objetivos es utilizar los recursos existentes en MultiPaquetería, por lo que se procede a cancelar un envío, se debe hacer esta cancelación en la base de datos central para que el sistema ya no le permita a la gente operativa relacionar el envío para intentar entregarlo. Por lo que el proceso de cancelación queda como se muestra a continuación:

En primer lugar nos abocaremos a las cancelaciones solicitadas por Unión del Oeste (103), para lo cual utilizamos un cursor que consulte aquellos envíos que se han solicitado cancelar y que no se han cancelado, como se muestra en la tabla 27.

```
Declare
cursor cCanc is
select a.numero_guia, b.civsituacion, b.rowid reng_guia, c.rowid reng_paquete
from uo_cancelaciones a, guia@prod b, paquete@prod c
where a.numero_guia = b.numero_guia
and a.numero_guia = c.numero_guia
and a.cancelado is null
and a.cancelacion_denegada is null;

verror_cod          varchar2(15);
verror_msg          varchar2(80);
```

Tabla 27. Cursor para detectar solicitud de nuevas cancelaciones.

Una vez identificados estos envíos, se consulta uno a uno la situación que tiene en la base de datos central, si el envío ya está entregado o está en proceso de entrega, se rechaza la solicitud de cancelación y se marca el registro como cancelación denegada. Pero si el envío no está en estas situaciones, se realiza una actualización en la base de datos central registrando la situación de que el envío ha sido

cancelado. Esta comparación de estatus y actualización de registro se muestra en la tabla 28.

```

Begin
  for vCanc in cCanc
  loop
    begin
      If vCanc.clvsituacion not in (6,7,1,3,20,109) then
        update guia@prod set
          clvsituacion = 20,
          fechacancelacion = sysdate,
          estaactiva = 0
          where rowid = vCanc.reng_guia;
        update paquete@prod set
          clvsituacion = 20
          where rowid = vCanc.reng_paquete;
        update uo_cancelaciones set
          fechacancelado = sysdate,
          cancelado = 'X'
          where numero_guia = vCanc.numero_guia;
        commit;
      else
        update uo_cancelaciones set
          cancelacion_denegada = 'X'
          where numero_guia = vCanc.numero_guia;
        commit;
      end if;
    exception
    when others then
      begin
        verror_cod := to_char(sqlcode);
        verror_msg := substr(sqlerrm,1,100);
        insert into uo_cancelaciones_probls (
          error,
          fecha_error,
          cancelacion_solicitada)
          values (substr(verror_cod||':'||verror_msg,1,100),
            sysdate,
            'X');
        commit;
      end;
    end;
  end loop;

```

Tabla 28. Actualización de estatus de registro a situación de cancelado por solicitud.

El siguiente tipo de cancelación es la cancelación por vigencia, cuyas características ya se describieron con anterioridad. Su proceso se define como sigue:

Lo primero es identificar los envíos que tienen mas de 16 días desde que se registró su documentación y que no han sido entregados o cancelados a través de un cursor, el cual se muestra en la tabla 29.

```
Declare
cursor cVenc is
select a.numeroguia,
to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr') dias_vigencia
from uo_regreso a, guia@prod b
where a.numeroguia = b.numeroguia
and to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr') >= 16
and b.clvsituacion not in (1, 7, 51, 57, 63, 20)
and a.vencido is null
group by a.numeroguia, to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr');
```

Tabla 29. Cursor para detectar envíos que se deben cancelar por vigencia.

Una vez identificados los envíos que tienen esta característica, se les da el mismo tratamiento que a las cancelaciones por solicitud, a diferencia que en este caso se agrega el comentario de 'ENVIO VENCIDO' para que posteriormente sea a través de este comentario que se identifique el tipo de cancelación que aplicó al envío, ya que realmente tanto los envíos cancelados por solicitud como los cancelados por vigencia guardan el mismo status en la base de datos central. Esta actualización se muestra en la tabla 30.

```

Begin
  for vVenc in cVenc
  loop
    begin
      update guia@prod set
        clvsituacion = 20,
        fechacancelacion = sysdate,
        estaactiva = 0,
        observaciones = 'ENVIO VENCIDO'
      where numeroguia = vVenc.numeroguia;
      update paquete@prod set
        clvsituacion = 20
      where numeroguia = vVenc.numeroguia;
      commit;
    exception
    when others then
      begin
        verror_cod := to_char(sqlcode);
        verror_msg := substr(sqlerrm,1,100);
        insert into uo_cancelaciones_probls (
          error,
          fecha_error,
          cancelacion_vencimiento)
          values (substr(verror_cod||' : '||verror_msg,1,100),
            sysdate,
            'X');
      end;
    end;
  end loop;
end;

```

Tabla 30. Actualización de estatus de envíos cancelados por vigencia.

El script completo para este tema se muestra en el Apéndice D relativo al Proceso de Cancelaciones.

#### 4.3.4 INFORME DE SITUACIONES A UNION DEL OESTE.

Las restantes tres actividades que debe realizar el servidor son las siguientes:

- Integrar e interpretar estas actualizaciones en su base de datos local.
- Generar el archivo que se transmitirá diariamente a Unión del Oeste, informando la situación actualizada de cada uno de sus envíos.
- Transmitir de manera automática el archivo hacia Unión del Oeste diariamente.

Como se mencionó anteriormente, al integrar los envíos en la operación normal de entrega, los cambios en la situación de los mismos será reflejada en la base de datos central, por lo es de esta base de datos de donde actualizaremos esta información en nuestra base de datos local a través del siguiente proceso:

Se define un cursor que obtiene la última situación de cada envío del proyecto mientras haya sufrido actualización en los últimos dos días. El modo de distinguirlos es a través de los primeros cuatro caracteres que forman el número de guía (7309).

```

Declare
cursor c_situacion is
select
a.referenciacliente
, a.numeroguia
, a.clvsituacion
, a.fechasituacion
, b.clvcausanoentrega
, a.fechaadocumentacionorigen
, a.fechaentrega
, a.nombrepersonarecibe
, substr(a.folioidentificacion,1,20) folioidentificacion
, substr(c.descripcion,1,30) descripcion
, decode(substr(upper(a.nombrepersonarecibe),1,10),null,null,substr(upper(d.nombre),1,10),1,2) consignatariooriginal
, a.observaciones
from guia@prod a
, paquete@prod b
, tiposidentificacion@prod c
, consignatario@prod d
where
a.numeroguia like '7309%'
and a.numeroguia = b.numeroguia
and a.clvtipoidentificacion = c.clvtipoidentificacion(+)
and a.clvconsignatario = d.clvconsignatario
and to_date(a.fechasituacion,'dd/mm/rr') > to_date(sysdate,'dd/mm/rr')-2;

```

Tabla 31. Identificación de envíos con cambios de estatus.

Como se puede observar en la tabla 31, se toman datos adicionales que permitirán interpretar la situación para informar de acuerdo a los estatus que se manejan para Unión del Oeste.

Posteriormente se realiza la inserción de las situaciones en la tabla local correspondiente. En caso de que la situación sea la misma y se haya generado en la misma fecha de la que se tiene como última registrada, el registro es automáticamente rechazado gracias a la llave primaria de la tabla. Este proceso de inserción se puede observar en la tabla 32.

```

Begin
  for v_situacion in c_situacion
  loop
    begin
      insert into uo_regreso(
        guia_uo,
        numeroguia,
        causanoentrega,
        situacion,
        fechasituacion,
        fechaentrega,
        fechain1,
        fechain2,
        nombrecibe,
        folioidentrecibe,
        identrecibe,
        consignatariooriginal,
        fechadocumentacion,
        vencido)
      values(
        v_situacion.referenciaciente,
        v_situacion.numeroguia,
        v_situacion.clvcausanoentrega,
        v_situacion.clvsituacion,
        v_situacion.fechasituacion,
        v_situacion.fechaentrega,
        decode(v_situacion.clvsituacion,102,v_situacion.fechasituacion,8,v_situacion.fechasituacion,null),
        decode(v_situacion.clvsituacion,103,v_situacion.fechasituacion,null),
        v_situacion.nombrepersonarecibe,
        v_situacion.folioidentificacion,
        v_situacion.descripcion,
        v_situacion.consignatariooriginal,
        v_situacion.fechadocumentacionorigen,
        decode(v_situacion.observaciones,'ENVIO VENCIDO','X',null)
      );
    end;
  end loop;
End;

```

Tabla 32. Inserción de nuevos estatus en la tabla correspondiente.

Posteriormente se toman los datos de la base de datos local y se interpretan desde el cursor a través de la utilización de la función decode<sup>16</sup>, como se observa en la tabla 33.

```

Declare
  VArchivo          utl_file.file_type;
  VArchftp          utl_file.file_type;
  vBuffer           varchar2(500);
  vCuenta           number(5) := 0;
  vTotEntEfe        number(5) := 0;
  vImpEntEfe        number(5) := 0;
  vTotEntChq        number(5) := 0;
  vImpEntChq        number(5) := 0;
  cursor c_genera is
  select
    a.rowid renglon,
    decode(a.situacion,
           1,100,
           7,100,
           51,100,
           57,100,
           63,100,
           13,decode(a.vencido,'X',106,102),
           20,decode(a.vencido,'X',106,102),
           35,decode(a.vencido,'X',106,102),
           53,decode(a.vencido,'X',106,102),
           62,decode(a.vencido,'X',106,102),
           101,decode(a.vencido,'X',106,102),
           104,decode(a.vencido,'X',106,102),
           106,decode(a.vencido,'X',106,102),
           107,decode(a.vencido,'X',106,102),
           5,103,
           34,103,
           58,103,
           8,decode(a.causanoentrega,7,103,5,103,104),
           102,104,
           103,105,
           10,106,
           37,106,
           108,106,
           101) situacion,
    a.guia_uo,
    b.numero_ident,
    b.transtype,
    b.datetime,
    b.amount_pesos,
    b.tipo_moneda,
    a.fechaent1,
    a.fechaent2,
    a.fechasituacion,
    a.fechaentrega,
    b.nombre_bnro,
    b.nombre2_bnro,
    b.direccion,
    b.colonia,
    b.ciudad,
    b.estado,
    b.prefijo,
    b.telefono,
    b.codigo_postal,
    b.remitente,

```

<sup>16</sup> Función que evalúa un dato y puede realizar dos cosas dependiendo si la evaluación es verdadera o falsa. Lleva la misma lógica que una sentencia IF.

```

a.folioidentrecibe,
a.identrecibe,
a.consignatariooriginal,
a.transmitido,
a.causanoentrega,
a.vencido
from uo_regreso a, uo_transf b
where a.guia_uo = b.numero_guia
and a.transmitido is null
order by situacion, b.transtype, a.guia_uo;

```

Tabla 33. Interpretación de estatus desde base de datos local.

Posteriormente se utiliza otro cursor para identificar solamente la última situación del envío ya que en la tabla se guarda una bitácora de todas las situaciones que vaya teniendo cada envío.

Ya identificada la última situación, se comienza a escribir registro a registro el archivo que se transmitirá a Unión del Oeste y se utilizan variables que van acumulando los datos que formarán el último registro del archivo (trailer).

Nótese en la tabla 34 que si la situación que se va a transmitir es Entregado o Cancelado, se actualiza la tabla para que no se vuelvan a transmitir los datos de ese envío.

Una vez que se ha generado el archivo a transmitir con todas las características mencionadas anteriormente, se procede a generar el archivo ftp que servirá para hacer la transmisión, como se observa en la tabla 35.

Tanto el proceso de cancelación como la generación de este archivo se controlarán a través del archivo bat mostrado en la tabla 36, que se ejecutará a través del scheduler de NT.

Nótese que se mandan ejecutar ambos scripts y después se utiliza un "ping" para establecer la comunicación con el equipo de Unión del Oeste, pero se utiliza como parámetro "-n 60", que lo que hace solicitar respuesta del equipo 60 veces independientemente de que la haya o no. Después de lo que tarda la ejecución del "ping" se manda la ejecución del archivo bat que a su vez ejecuta al ftp para la transmisión. Esta pequeña espera asegura que al momento de intentar hacer el ftp, los equipos no tengan problemas de comunicación.

```

VArchivo := utl_file.fopen('c:\uo\Regresouo',mu||to_char(sysdate,'mmdd')||'01.txt','w');
For v_genera in c_genera
Loop
  Declare
  cursor c_ultima_sit is
  select
    guia_uo,
    max(fechasituacion) maxfechasit
  from uo_regreso
  where guia_uo = v_genera.guia_uo
  group by guia_uo;
  v_guia          varchar2(15);
  v_fecha         date;

  begin
    open c_ultima_sit;
    fetch c_ultima_sit into v_guia, v_fecha;
    close c_ultima_sit;
    if v_genera.guia_uo = v_guia and v_genera.fechasituacion = v_fecha and v_genera.transmitido is null then
      vBuffer := rpad(''||to_char(v_genera.situacion)
        ||'|'||v_genera.guia_uo
        ||'|'||v_genera.numero_ident
        ||'|'||v_genera.transtype
        ||'|'||to_char(v_genera.datetime,'dd-mm-yyyy hh24:mi')
        ||'|'||to_char(v_genera.amount_pesos)
        ||'|00.'||v_genera.tipo_moneda
        ||'|'||to_char(v_genera.fechaint1,'dd-mm-yyyy hh24:mi')
        ||'|'||to_char(v_genera.fechaint2,'dd-mm-yyyy hh24:mi')
        ||'|'||to_char(v_genera.fechaentrega,'dd-mm-yyyy hh24:mi')
        ||'|'||v_genera.nombre_bvro
        ||'|'||v_genera.nombre2_bvro
        ||'|'||v_genera.direccion
        ||'|'||v_genera.colonia
        ||'|'||v_genera.ciudad
        ||'|'||v_genera.estado
        ||'|'||v_genera.prefijo
        ||'|'||v_genera.telefono
        ||'|'||v_genera.codigo_postal
        ||'|'||v_genera.remitente
        ||'|'||v_genera.folioidentrecibe
        ||'|'||v_genera.identrecibe
        ||'|'||to_char(v_genera.consignatariooriginal)
        ||'|500.'');
      utl_file.put_line(vArchivo,vBuffer);
      if v_genera.situacion = 100 then
        if v_genera.transtype = 'EFE' then
          vTotEntEfe := vTotEntEfe + 1;
          vImpEntEfe := vImpEntEfe + v_genera.amount_pesos;
        else
          vTotEntChq := vTotEntChq + 1;
          vImpEntChq := vImpEntChq + v_genera.amount_pesos;
        end if;
      end if;
      if v_genera.situacion in (100, 102, 106) then
        update uo_regreso set transmitido = 'X' where guia_uo = v_genera.guia_uo;
      end if;
      vCuenta := vCuenta + 1;
    end if;
  exception
  when others then
    insert into errores valores ('no escribi en archivo el dia'||to_char(sysdate,'ddmmyy'));
  end;
end loop;
vBuffer := rpad('||120'||to_char(vCuenta)||'|'||to_char(sysdate,'dd-mm-yyyy
hh24:mi')||'|'||to_char(vImpEntEfe)||'.00'||to_char(vTotEntEfe)||'|'||to_char(vImpEntChq)||'.00'||to_char(vTotEntChq)||'|'
||500.'');
utl_file.put_line(vArchivo,vBuffer);

```

Tabla 34. Identificación de última situación y escritura del archivo a transmitir.

```

VArchftp := utl_file.fopen('c:\uo\bats','transfer.ftp', 'w');
VBuffer := 'open equipo_uo';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'multipaq';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'multipaq';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'bin';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'prompt off';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'cd in';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'cd \uo\regresouo';
Utl_file.put_line(vArchftp,vBuffer);
VBuffer := 'put mu||to_char(sysdate,'mmdd')||'01.txt';
Utl_file.put_line(vArchftp,vBuffer);
if utl_file.is_open(vArchftp) then
    utl_file.fclose(vArchftp);
end if;

```

Tabla 35. Creación de archivo bat para ejecutar la transmisión del archivo generado.

```

Sqlplus -s uniono/ uniono @ c:\uo\scripts\uo_cancelaciones
Sqlplus -s uniono / uniono @ c:\uo\scripts\uo_regreso
Ping equipo_uo -n 60
Call c:\uo\scripts\enlace_ftp

```

Tabla 36. Archivo bat que se ejecutará con el Scheduler de NT.

El script completo para esta parte del desarrollo se ilustra en el Apéndice E relativo al Proceso para Informar situaciones.

## **CONCLUSIONES.**

Con el desarrollo del presente Trabajo de Seminario se concreta satisfactoriamente un proyecto que forma parte de la operación de una empresa con una gran estructura organizacional como lo es MultiPaquetería y también se pudieron contemplar los aspectos relacionados a los Sistemas de Información, objetivo del Seminario de Titulación.

Se logró mostrar que dos empresas de giros diferentes pueden llegar a interactuar operativa e informativamente para generar estrategias comerciales que les generen beneficios mutuos, y para esto fue necesario e importante el análisis de todos los procesos que involucran el proyecto y más aun el modo en que la parte conceptual de las políticas del negocio pudieran ser plasmadas y controladas desde conceptos de Sistemas de Información.

En términos de Sistemas de Información se ha logrado implementar un esquema que contempla la comunicación entre ambas empresas y un flujo completo de la información que gira en torno al proyecto, utilizando recursos de hardware, software, manejadores de bases de datos y telecomunicaciones que permitieron una adecuada incorporación de la información generada por el proyecto a las bases de datos productivas de MultiPaquetería y por consiguiente a su operación.

Con la culminación de este proyecto, se logró que MultiPaquetería concrete una nueva alianza comercial que la reitera como una de las empresas nacionales de mensajería y paquetería con grandes capacidades informáticas, gracias a las que hoy por hoy es una empresa líder en su ramo.

## **APENDICE.**

## A.Sobre la creación de la base de datos.

```
.....
--** CREACION Y MODIFICACION DE LA BASE DE DATOS **
.....

--** genera un log del script
spool crear_bd.log

connect internal/password

--** desactivar segmentos de rollback predeterminados
alter rollback segment rb1 offline;
alter rollback segment rb2 offline;
alter rollback segment rb3 offline;
alter rollback segment rb4 offline;
alter rollback segment rb5 offline;
alter rollback segment rb6 offline;
alter rollback segment rb7 offline;
alter rollback segment rb8 offline;

--** MANIPULACION Y CREACION DE TABLESPACES

--** borra tablespace default para segmentos de rollback
drop tablespace rollback_data including contents;
hos del d:\oracle\oradata\unio\RBs1.orcl.ora
--** modifica el tamaño del tablespace del usuario system
alter database
datafile 'd:\oracle\oradata\unio\Sys1.orcl.ora' resize 30m;
--** crea tablespace utilizado como temporal para el usuario unio
create tablespace tempo
datafile 'd:\oracle\oradata\unio\tempo01.dbf' size 100m;
--** crea un nuevo tablespace para los segmentos de rollback
create tablespace rbs
datafile 'd:\oracle\oradata\unio\rbs01.dbf' size 100m;
--** crea tablespace para objetos del usuario unio
create tablespace unioeste
datafile 'd:\oracle\oradata\unio\unio01.dbf' size 250m;
--** crea tablespace para índices de tablas del usuario unio
create tablespace indices_unioeste
datafile 'd:\oracle\oradata\unio\ind_unio01.dbf' size 100m;

--** CREA NUEVOS SEGMENTOS DE ROLLBACK
create public rollback segment r01
storage (initial 150k next 50k)
tablespace rbs;
create public rollback segment r02
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r03
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r04
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r05
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r06
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r07
storage (initial 50k next 50k)
tablespace rbs;
```

```

create public rollback segment r08
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r09
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r10
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r11
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r12
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r13
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r14
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r15
storage (initial 50k next 50)
tablespace rbs;
create public rollback segment r16
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r17
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r18
storage (initial 50k next 50k)
tablespace rbs;
create public rollback segment r19
storage (initial 50k next 50)
tablespace rbs;
create public rollback segment r20
storage (initial 50k next 50k)
tablespace rbs;

```

--\*\* ACTIVA NUEVOS SEGMENTOS DE ROLLBACK

```

alter rollback segment r01 online;
alter rollback segment r02 online;
alter rollback segment r03 online;
alter rollback segment r04 online;
alter rollback segment r05 online;
alter rollback segment r06 online;
alter rollback segment r07 online;
alter rollback segment r08 online;
alter rollback segment r09 online;
alter rollback segment r10 online;
alter rollback segment r11 online;
alter rollback segment r12 online;
alter rollback segment r13 online;
alter rollback segment r14 online;
alter rollback segment r15 online;
alter rollback segment r16 online;
alter rollback segment r17 online;
alter rollback segment r18 online;
alter rollback segment r19 online;
alter rollback segment r20 online;

```

connect system/system

--\*\* CREACION DEL USUARIO UNIONO

```

create user uniono
identified by uniono
default tablespace unionoeste
temporary tablespace tempo
quota unlimited on unionoeste

```

```
quota unlimited on indices_unioeste  
quota unlimited on tempo;
```

```
--** OTORGA PRIVILEGIOS PARA CONECTARSE Y GENERAR  
--** OBJETOS PROPIOS AL USUARIO UNIONO  
grant connect, resource to unioeste;
```

```
--** CIERRA EL LOG DEL SCRIPT  
spool off;
```

```
exit;
```

## B. Creación de tablas del usuario.

```

DROP TABLE ERRORES;
DROP TABLE UO_CODIGO_IATA;
DROP TABLE UO_ERROR;
DROP TABLE UO_PROBLS;
DROP TABLE UO_REGRESO;
DROP TABLE UO_TRANSF;
DROP TABLE UO_CANCELACIONES;
DROP TABLE UO_CANCELACIONES_PROBLS;

CREATE TABLE UO_REGRESO (
  GUIA_UO          VARCHAR2(15),
  NUMEROGUIA      VARCHAR2(15)  NOT NULL,
  CAUSANOENTREGA  NUMBER(3),
  SITUACION       NUMBER(3)    NOT NULL,
  FECHASITUACION  DATE         NOT NULL,
  FECHAENTREGA    DATE,
  FECHAI1        DATE,
  FECHAI2        DATE,
  NOMBRERECIBE   VARCHAR2(40),
  FOLIOIDENTRECIBE VARCHAR2(20),
  IDENTRECIBE    VARCHAR2(30),
  CONSIGNATARIOORIGINAL NUMBER(1),
  TRANSMITIDO    VARCHAR2(1),
  FECHADOCUMENTACION DATE,
  VENCIDO        VARCHAR2(1));
ALTER TABLE UO_REGRESO ADD CONSTRAINT PK_UO_REGRESO PRIMARY KEY (NUMEROGUIA, SITUACION, FECHASITUACION);

CREATE TABLE UO_TRANSF (
  TIPO_REGISTRO   VARCHAR2(3),
  NUMERO_GUIA     VARCHAR2(15)  NOT NULL,
  NUMERO_IDENT    VARCHAR2(12),
  TRANSTYPE      VARCHAR2(3),
  ORIGIN         VARCHAR2(3),
  DATETIME       DATE,
  AMOUNT_PESOS   NUMBER(9,2),
  REMITENTE      VARCHAR2(40),
  PREFIJO RTE    VARCHAR2(7),
  NUMERO RTE     VARCHAR2(15),
  NOMBRE_BNRO    VARCHAR2(40),
  NOMBRE2_BNRO   VARCHAR2(40),
  DIRECCION      VARCHAR2(40),
  COLONIA        VARCHAR2(30),
  CIUDAD         VARCHAR2(30),
  ESTADO         VARCHAR2(3),
  PREFIJO        VARCHAR2(7),
  TELEFONO       VARCHAR2(15),
  CODIGO_POSTAL  VARCHAR2(5),
  TIPO_MONEDA    VARCHAR2(3),
  FECHA_TRANS    DATE,
  ARCHIVO_ORIGEN VARCHAR2(12),
  CP_VALIDO      VARCHAR2(1));
ALTER TABLE UO_TRANSF ADD CONSTRAINT PK_UO_TRANSF PRIMARY KEY (NUMERO_GUIA);

CREATE TABLE UO_CANCELACIONES (
  TIPO_REGISTRO   VARCHAR2(3),
  NUMERO_GUIA     VARCHAR2(15)  NOT NULL,
  DATETIME       DATE,
  FECHA_TRANS    DATE,
  FECHACANCELADO DATE,
  CANCELADO      VARCHAR2(1),
  CANCELACION_DENEGADA VARCHAR2(1),

```

```

ARCHIVO_ORIGEN          VARCHAR2(12);
ALTER TABLE UO_CANCELACIONES ADD CONSTRAINT PK_UO_CANCELACIONES PRIMARY KEY (NUMERO_GUIA);

```

```

CREATE TABLE UO_PROBLS (
TIPO_REGISTRO          VARCHAR2(3),
NUMERO_GUIA           VARCHAR2(15),
NUMERO_IDENT          VARCHAR2(12),
TRANSTYPE             VARCHAR2(3),
ORIGIN                VARCHAR2(3),
DATETIME              VARCHAR2(16),
AMOUNT_PESOS          VARCHAR2(12),
REMITENTE             VARCHAR2(40),
PREFIJO_RTE           VARCHAR2(7),
NUMERO_RTE            VARCHAR2(7),
NOMBRE_BNRO           VARCHAR2(40),
NOMBRE2_BNRO          VARCHAR2(40),
DIRECCION             VARCHAR2(40),
COLONIA               VARCHAR2(30),
CIUDAD                VARCHAR2(30),
ESTADO                VARCHAR2(3),
PREFIJO               VARCHAR2(7),
TELEFONO              VARCHAR2(7),
CODIGO_POSTAL         VARCHAR2(5),
TIPO_MONEDA           VARCHAR2(3),
ERROR                 VARCHAR2(150),
FECHA_TRANS           DATE,
ARCHIVO_ORIGEN        VARCHAR2(12));

```

```

CREATE TABLE ERRORES (
ERROR                 VARCHAR2(50));

```

```

CREATE TABLE UO_CODIGO_IATA (
CODIGO_POSTAL         NUMBER(5),
ASENTAMIENTO_HUMANO  VARCHAR2(40) NOT NULL,
ESTADO                VARCHAR2(3),
MUNICIPIO             VARCHAR2(3),
REGION                VARCHAR2(3),
CABECERA              VARCHAR2(20),
CLAVE_IATA            VARCHAR2(3));

```

```

CREATE TABLE UO_ERROR (
FECHA_ERROR           DATE,
ERROR                 VARCHAR2(100),
ARCHIVO_ORIGEN        VARCHAR2(12));

```

```

CREATE TABLE UO_CANCELACIONES_PROBLS (
ERROR                 VARCHAR2(100),
FECHA_ERROR           DATE(12),
CANCELACION_SOLICITADA VARCHAR2(1),
CANCELACION_VENCIMIENTO VARCHAR2(1));

```

```

CREATE TABLE ARCHIVOS_PROCESO (
NOMBRE_ARCHIVO        VARCHAR2(12),
PROCESADO              VARCHAR2(1),
FECHA_PROCESO         DATE);

```

```

CREATE TABLE UO_REG_PROCESA (
REGION                VARCHAR2(3),
REG_PROCESA           VARCHAR2(3));
ALTER TABLE UO_REG_PROCESA ADD CONSTRAINT PK_REG_PROCESA PRIMARY KEY (REGION);

```

## C. Proceso de incorporación y distribución de Información.

```

SPOOL CAUO\LOGS\REGIONES.LOG
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') from dual;
DECLARE
vArchivo                UTL_FILE.FILE_TYPE,
vArchExis              UTL_FILE.FILE_TYPE,
vArchCons              UTL_FILE.FILE_TYPE,
vArchValid             UTL_FILE.FILE_TYPE;

vTipo_Registro         varchar2(3);
vNumero_Guia          varchar2(15);
vNumero_Ident         varchar2(12);
vTranstype            varchar2(3);
vOrigin               varchar2(3);
vDatetime             varchar2(16);
vAmount_Pesos        varchar2(12);
vRemitente            varchar2(40);
vPrefijo_Rte          varchar2(7);
vNumero_Rte           varchar2(15);
vNombre_Buro          varchar2(40);
vNombre2_Buro         varchar2(40);
vDireccion            varchar2(40);
vColonia              varchar2(30);
vCiudad               varchar2(30);
vEstado               varchar2(3);
vPrefijo              varchar2(7);
vTelefono             varchar2(15);
vCodigo_Postal        varchar2(5);
vTipo_Moneda          varchar2(3);

vCount_Arch           number(1);
vNombre_Arch          varchar2(12);
vBuffer               varchar2(500);
verror_cod            varchar2(15);
verror_msg            varchar2(80);
v_cuenta              number := 0;
v_guarda              number := 1;
v_totereg             number(5);
v_totefe              number(5);
v_totchq              number(5);
v_impefe              number(10);
v_impchq              number(10);
v_iniproc             number(5);
v_finproc             number(5);
v_numarch             varchar2(2);

BEGIN
if upper(to_char(sysdate,'dy')) = 'SÁB' then
  if to_number(to_char(sysdate,'hh24')) <= 8 then
    v_iniproc := 1;
    v_finproc := 1;
    v_numarch := '01';
  else
    v_iniproc := 2;
    v_finproc := 2;
    v_numarch := '02';
  end if;
else
  if to_number(to_char(sysdate,'hh24')) <= 15 then
    v_iniproc := 1;
    v_finproc := 2;
    v_numarch := '01';
  else
    v_iniproc := 3;
    v_finproc := 3;
    v_numarch := '02';
  end if;
end if;
end if;

```

```

FOR vCount_Arch in v_iniproc .. v_finproc LOOP
BEGIN
  vNombre_Arch := 'WM'[to_char(sysdate,'mmdd')]||'0'[to_char(vCount_Arch)]||'.TXT';
  insert into archivos_proceso values (vNombre_Arch, null,null);
EXCEPTION
  when others then null;
END;
END LOOP;
DECLARE
  cursor cArchProc is
  select nombre_archivo from archivos_proceso where procesado is null;
BEGIN
  for vArchProc in cArchProc loop
  BEGIN
    v_totreg      := 0;
    v_totefe      := 0;
    v_totchq      := 0;
    v_impefe      := 0;
    v_impchq      := 0;
    vArchivo      := UTL_FILE.FOPEN ('d:\UnionO',vArchProc.nombre_archivo,'r');
    LOOP
      BEGIN
        UTL_FILE.GET_LINE(vArchivo,vBuffer);
        vTipo_Registro := SUBSTR(vBuffer, 2,
INSTR(vBuffer,'1,1) - 3);
        vNumero_Guia   := SUBSTR(vBuffer, INSTR(vBuffer,'1,1) + 2, INSTR(vBuffer,'1,2) - INSTR(vBuffer,'1,1) - 3);
        vNumero_Ident  := SUBSTR(vBuffer, INSTR(vBuffer,'1,2) + 2, INSTR(vBuffer,'1,3) - INSTR(vBuffer,'1,2) - 3);
        vTranstype     := SUBSTR(vBuffer, INSTR(vBuffer,'1,3) + 2, INSTR(vBuffer,'1,4) - INSTR(vBuffer,'1,3) - 3);
        vOrigin        := SUBSTR(vBuffer, INSTR(vBuffer,'1,4) + 2, INSTR(vBuffer,'1,5) - INSTR(vBuffer,'1,4) - 3);
        vDatetime      := SUBSTR(vBuffer, INSTR(vBuffer,'1,5) + 2, INSTR(vBuffer,'1,6) - INSTR(vBuffer,'1,5) - 3);
        vAmount_Pesos  := SUBSTR(vBuffer, INSTR(vBuffer,'1,6) + 1, INSTR(vBuffer,'1,7) - INSTR(vBuffer,'1,6) - 1);
        vRemitente     := SUBSTR(vBuffer, INSTR(vBuffer,'1,7) + 2, INSTR(vBuffer,'1,8) - INSTR(vBuffer,'1,7) - 3);
        vPrefijo_Rte   := SUBSTR(vBuffer, INSTR(vBuffer,'1,8) + 2, INSTR(vBuffer,'1,9) - INSTR(vBuffer,'1,8) - 3);
        vNumero_Rte    := SUBSTR(vBuffer, INSTR(vBuffer,'1,9) + 2, INSTR(vBuffer,'1,10) - INSTR(vBuffer,'1,9) - 3);
        vNombre_Bnro   := SUBSTR(vBuffer, INSTR(vBuffer,'1,10) + 2, INSTR(vBuffer,'1,11) - INSTR(vBuffer,'1,10) - 3);
        vNombre2_Bnro  := SUBSTR(vBuffer, INSTR(vBuffer,'1,11) + 2, INSTR(vBuffer,'1,12) - INSTR(vBuffer,'1,11) - 3);
        vDireccion     := SUBSTR(vBuffer, INSTR(vBuffer,'1,12) + 2, INSTR(vBuffer,'1,13) - INSTR(vBuffer,'1,12) - 3);
        vColonia       := SUBSTR(vBuffer, INSTR(vBuffer,'1,13) + 2, INSTR(vBuffer,'1,14) - INSTR(vBuffer,'1,13) - 3);
        vCiudad        := SUBSTR(vBuffer, INSTR(vBuffer,'1,14) + 2, INSTR(vBuffer,'1,15) - INSTR(vBuffer,'1,14) - 3);
        vEstado        := SUBSTR(vBuffer, INSTR(vBuffer,'1,15) + 2, INSTR(vBuffer,'1,16) - INSTR(vBuffer,'1,15) - 3);
        vPrefijo       := SUBSTR(vBuffer, INSTR(vBuffer,'1,16) + 2, INSTR(vBuffer,'1,17) - INSTR(vBuffer,'1,16) - 3);
        vTelefono      := SUBSTR(vBuffer, INSTR(vBuffer,'1,17) + 2, INSTR(vBuffer,'1,18) - INSTR(vBuffer,'1,17) - 3);
        vCodigo_Postal := SUBSTR(vBuffer, INSTR(vBuffer,'1,18) + 2, INSTR(vBuffer,'1,19) - INSTR(vBuffer,'1,18) - 3);
        vTipo_Moneda   := SUBSTR(vBuffer, INSTR(vBuffer,'1,19) + 2, 3);
        if vTipo_Registro in ('110','111') then
          v_totreg := v_totreg + 1;
          if vTipo_Registro = '110' then
            if vTranstype = 'EFE' then
              v_totefe := v_totefe + 1;
              v_impefe := v_impefe + to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,'1,1)-1));
            else
              v_totchq := v_totchq + 1;
              v_impchq := v_impchq + to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,'1,1)-1));
            end if;
            insert into uo_transf (tipo_registro, numero_guia, numero_ident, transtype,
              origin, datetime, amount_pesos, remitente, prefijo_rte,
              numero_rte, nombre_bnro, nombre2_bnro, direccion, colonia,
              ciudad, estado, prefijo, telefono, codigo_postal,
              tipo_moneda, fecha_trans, archivo_origen
            )
              values (vTipo_Registro, vNumero_Guia, vNumero_Ident, vTranstype, vOrigin,
                to_date(vDatetime, 'dd-mm-yyyy hh24:mi'),
                to_number(substr(vAmount_Pesos,1,instr(vAmount_Pesos,'1,1)-1)),
                vRemitente, vPrefijo_Rte, vNumero_Rte, vNombre_Bnro, vNombre2_Bnro,
                vDireccion, vColonia, vCiudad, vEstado, vPrefijo, vTelefono,
                vCodigo_Postal, vTipo_Moneda, sysdate, vNombre_Arch
              );
          elsif vTipo_Registro = '111' then
            insert into uo_cancelaciones (tipo_registro, numero_guia, datetime,
              fecha_trans, archivo_origen

```

```

        )
        values (vTipo_Registro, vNumero_Guia,
              to_date(vDatetime, 'dd-mm-yyyy hh24:mi'), sysdate, vNombre_Arch
              );
    end if;
    if vTipo_Registro = '120' then
    if to_number(vNumero_Guia) <> v_totreg or
    to_number(substr(vRemitente, 1, instr(vRemitente, ',', 1, 1) - 1)) <> v_impefe or
    to_number(vPrefijo_Rte) <> v_totefe or
    to_number(vNumero_Rte) <> v_totchq or
    to_number(substr(vNombre_Bnro, 1, instr(vNombre_Bnro, ',', 1, 1) - 1)) <> v_impchq then
    delete uo_transf where archivo_origen = vNombre_Arch;
    commit;
    IF NOT UTL_FILE.IS_OPEN(vArchCons) THEN
    VArchCons := UTL_FILE.FOPEN ('c:\uo\Inconsis', 'IC' || to_char(sysdate, 'mmdd') || v_numarch || '.TXT', 'w');
    END IF;
    VBuffer := vNombre_Arch;
    UTL_FILE.PUT_LINE(vArchCons, vBuffer);
    end if;
    end if;
exception
when no_data_found then
exit;
when others then
begin
    VERROR_COD := to_char(sqlcode);
    VERROR_MSG := substr(sqlerrm, 1, 80);
    Insert into uo_probls (tipo_registro, numero_guia, numero_ident,
                        transtype, origin, datetime, amount_pesos,
                        remitente, prefijo_rte, numero_rte, nombre_bnro,
                        nombre2_bnro, direccion, colonia, ciudad, estado,
                        prefijo, telefono, codigo_postal, tipo_moneda,
                        error, fecha_trans, archivo_origen
                        )
    values (vTipo_Registro, vNumero_Guia, vNumero_Ident, vTranstype, vOrigin,
          vDatetime, vAmount_Pesos, vRemitente, vPrefijo_Rte, vNumero_Rte,
          vNombre_Bnro, vNombre2_Bnro, vDireccion, vColonia, vCiudad, vEstado,
          vPrefijo, vTelefono, vCodigo_Postal, vTipo_Moneda,
          substr(verror_cod) : '||verror_msg, 1, 80), sysdate, vNombre_Arch
          );
    exception
    when others then null;
    end;
end;
if v_cuenta > v_guarda then
v_cuenta := 1;
commit;
else
v_cuenta := v_cuenta + 1;
end if;
end loop;
if utl_file.is_open(vArchivo) then
utl_file.fclose (vArchivo);
end if;
update archivos_proceso
set procesado = 'X',
fecha_proceso = sysdate
where nombre_archivo = vArchProc.nombre_archivo;
commit;
exception
when utl_file.read_error or utl_file.invalid_filehandle or utl_file.invalid_operation or utl_file.internal_error then
if not utl_file.is_open(vArchExis) then
vArchExis := UTL_FILE.FOPEN ('c:\uo\Existe', 'EX' || to_char(sysdate, 'mmdd') || v_numarch || '.TXT', 'w');
end if;
when others then null;
end;
end loop;
if utl_file.is_open(vArchExis) then
utl_file.fclose (vArchExis);
end if;

```

```

if utl_file.is_open(vArchCons) then
    utl_file.fclose (vArchCons);
end if;
vArchValid := utl_file.fopen ('c:\uo\Bats','TransErr.ftp','w');
vBuffer := 'open equipo_uo';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'multipaq';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'multipaq';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'bin';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'prompt off';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'cd in';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'lcd \uo\existe';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'put ex||to_char(sysdate,'mmdd')||v_numarch||'.txt';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'lcd \uo\inconsis';
utl_file.put_line(vArchValid,vBuffer);
vBuffer := 'put ic||to_char(sysdate,'mmdd')||v_numarch||'.txt';
utl_file.put_line(vArchValid,vBuffer);
if utl_file.is_open(vArchValid) then
    utl_file.fclose(vArchValid);
end if;

declare
cursor cREG is
select a.region, b.reg_procesa from uo_codigo_iata a, uo_reg_procesa b
where a.region = b.region
group by a.region, b.reg_procesa;
vFECHA      DATE      := SYSDATE;
vArcMain    UTL_FILE.FILE_TYPE;
vBufMain    varchar2(200);
vArcMasa    UTL_FILE.FILE_TYPE;
vBufMasa    varchar2(350);
vArchBat    UTL_FILE.FILE_TYPE;
vBufbat     varchar2(200);
vCMasiva    number(6) := 2;
BEGIN
vArcMasa := UTL_FILE.FOPEN ('c:\uo\Masiva','uo'||to_char(sysdate,'ddmm')||v_numarch||'.txt' , 'w');
vBufMasa := 'H000001 73587309 uo'||to_char(sysdate,'ddmmyy');
UTL_FILE.PUT_LINE(vArcMasa,vBufMasa);
vArcMain := UTL_FILE.FOPEN ('c:\uo\bats','MainBat.bat' , 'w');
vBufMain := 'echo Transfiriendo archivos a Regiones';
UTL_FILE.PUT_LINE(vArcMain,vBufMain);
FOR vREG IN cREG LOOP
    BEGIN
        DECLARE
            CURSOR cHayReg IS
                SELECT COUNT(*) FROM uo_TRANSF
                WHERE CODIGO_POSTAL IN (SELECT CODIGO_POSTAL
                FROM uo_CODIGO_IATA
                WHERE REGION = vREG.REGION)
                AND FECHA_TRANS > (sysdate - 0.125);
            CURSOR cSORT IS
                SELECT
                    A.TIPO_REGISTRO,
                    A.NUMERO_GUIA,
                    A.NUMERO_IDENT,
                    A.TRANSTYPE,
                    A.ORIGIN,
                    A.DATETIME,
                    A.AMOUNT_PESOS,
                    A.REMITENTE,
                    A.PREFIJO_RTE,
                    A.NUMERO_RTE,
                    A.NOMBRE_BNRO,

```

```

A.NOMBRE2_BNRO,
A.DIRECCION,
A.COLONIA,
A.CIUDAD,
A.ESTADO,
A.PREFIJO,
A.TELEFONO,
A.CODIGO_POSTAL,
A.TIPO_MONEDA,
B.CLAVE_IATA
FROM uo_TRANSF A, uo_CODIGO_IATA B
WHERE A.CODIGO_POSTAL = B.CODIGO_POSTAL AND
      A.CODIGO_POSTAL IN (SELECT CODIGO_POSTAL
                          FROM uo_CODIGO_IATA
                          WHERE REGION = vREG.REGION)
AND FECHA_TRANS > (sysdate - 0.125)
GROUP BY A.TIPO_REGISTRO,
         A.NUMERO_GUIA,
         A.NUMERO_IDENT,
         A.TRANSTYPE,
         A.ORIGIN,
         A.DATETIME,
         A.AMOUNT_PESOS,
         A.REMITENTE,
         A.PREFIJO_RTE,
         A.NUMERO_RTE,
         A.NOMBRE_BNRO,
         A.NOMBRE2_BNRO,
         A.DIRECCION,
         A.COLONIA,
         A.CIUDAD,
         A.ESTADO,
         A.PREFIJO,
         A.TELEFONO,
         A.CODIGO_POSTAL,
         A.TIPO_MONEDA,
         B.CLAVE_IATA;
vTipo_Registro1      varchar2(3);
vNumero_Guia1       varchar2(15);
vNumero_Ident1      varchar2(12);
vTranstype1         varchar2(3);
vOrigin1            varchar2(3);
vDatetime1          varchar2(16);
vAmount_Pesos1     varchar2(12);
vRemitente1         varchar2(40);
vPrefijo_Rte1       varchar2(7);
vNumero_Rte1        varchar2(15);
vNombre_Bnro1       varchar2(40);
vNombre2_Bnro1      varchar2(40);
vDireccion1         varchar2(40);
vColonial1          varchar2(30);
vCiudad1            varchar2(30);
vEstado1            varchar2(3);
vPrefijo1           varchar2(7);
vTelefono1          varchar2(15);
vCodigo_Postal1     varchar2(5);
vTipo_Monedas1      varchar2(3);
vIata1              varchar2(3);
vBuffer1            varchar2(325);
vCheques            number(5)      := 0;
vImpCheques         number(9,2)   := 0;
vEfectivos          number(5)      := 0;
vImpEfectivos       number(9,2)   := 0;
vRegistros          number(5)      := 0;
vFec                date;
vArchReg            UTL_FILE.FILE_TYPE;
vHayDatos           number(7);
BEGIN
  Open cHayReg;
  fetch cHayReg into vHayDatos;

```

```

Close chayReg;
If vHayDatos > 0 then
vArchReg := UTL_FILE.FOPEN ('c:\uo\regiones\|vREG.REGION,'WM'|to_char(sysdate,'mmdd')||v_numarch||'.TXT','w');
FOR vSORT IN cSORT LOOP
BEGIN
  if vSORT.TRANSTYPE = 'EFE' then
    vEfectivos := vEfectivos + 1;
    vImpEfectivos := vImpEfectivos + vSORT.AMOUNT_PESOS;
  else
    vCheques := vCheques + 1;
    vImpCheques := vImpCheques + vSORT.AMOUNT_PESOS;
  end if;
vRegistros := vRegistros + 1;
vTipo_Registro1 := vSORT.Tipo_Registro;
vNumero_Guia1 := vSORT.Numero_Guia;
vNumero_Ident1 := vSORT.Numero_Ident;
vTranstype1 := vSORT.Transtype;
vOrigin1 := vSORT.Origin;
vDatetime1 := to_char(vSORT.Datetime,'dd-mm-yyyy hh24:mi');
vAmount_Pesos1 := to_char(vSORT.Amount_Pesos);
vRemitente1 := vSORT.Remitente;
vPrefijo_Rte1 := vSORT.Prefijo_Rte;
vNumero_Rte1 := vSORT.Numero_Rte;
vNombre_Bnro1 := vSORT.Nombre_Bnro;
vNombre2_Bnro1 := vSORT.Nombre2_Bnro;
vDireccion1 := vSORT.Direccion;
vColonial1 := vSORT.Colonia;
vCiudad1 := vSORT.Ciudad;
vEstado1 := vSORT.Estado;
vPrefijo1 := vSORT.Prefijo;
vTelefono1 := vSORT.Telefono;
vCodigo_Postal1 := vSORT.Codigo_Postal;
vTipo_Moneda1 := vSORT.Tipo_Moneda;
vlata1 := vSORT.Clave_lata;
vBuffer1 := ""||vTipo_Registro1||"";
vNumero_Guia1||"";
vNumero_Ident1||"";
vTranstype1||"";
vOrigin1||"";
vDatetime1||"";
vAmount_Pesos1||'.00.';
vRemitente1||"";
vPrefijo_Rte1||"";
vNumero_Rte1||"";
vNombre_Bnro1||"";
vNombre2_Bnro1||"";
vDireccion1||"";
vColonial1||"";
vCiudad1||"";
vEstado1||"";
vPrefijo1||"";
vTelefono1||"";
vCodigo_Postal1||"";
vTipo_Moneda1||"";
UTL_FILE.PUT_LINE(vArchReg,vBuffer1);
update uo_transf set cp_valido = 'X'
where numero_guia = vNumero_Guia1;
commit;
vFEC := vSORT.DATE TIME;
vBufMasa := 'D'||pad(to_char(vCMasiva),6,'0')||rpad(trim(substr(vNombre_Bnro1||vNombre2_Bnro1,1,40)),40,' ')||
pad(nvl(vCodigo_Postal1,'0'),5,' ')||rpad(' ',13,' ')||rpad(trim(substr(nvl(vDireccion1,'-'),1,35)),35,' ')||
rpad(trim(substr(nvl(vColonial1,'-'),1,30)),30,' ')||rpad(trim(substr(nvl(vTelefono1,'0'),1,15)),15,' ')||
rpad(trim(substr(nvl(vCiudad1,'-'),1,40)),40,' ')||vlata1||rpad(' ',50,' ')||pad('0.00','8',' ')||
' uo|to_char(sysdate,'ddmmyy')||rpad(vNumero_Guia1,20,' ')||pad('0','6',' ')||
pad('0','6',' ')||pad('0','6',' ')||pad('0','7',' ')||'0';
UTL_FILE.PUT_LINE(vArchReg,vBufMasa);
vCMasiva := vCMasiva + 1;
END;
END LOOP;
vBuffer1 := "120"|| substr('000000000',1,(10-length(TO_CHAR(vRegistros)))||TO_CHAR(vRegistros) ||

```



```
EXCEPTION
WHEN OTHERS THEN
  verror_cod      := TO_CHAR(SQLCODE);
  verror_msg      := SUBSTR(SQLERRM,1,80);
  INSERT INTO UO_ERROR (FECHA_ERROR, ERROR, ARCHIVO_ORIGEN)
  VALUES (sysdate, verror_cod||' '||verror_msg, vNombre_Arch);
  COMMIT;
END;
END;
/
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') from dual;
SPOOL OFF
EXIT
```

## D. Proceso de Cancelaciones.

```
spool c:\uol\logs\cancelaciones.log
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') from dual;
declare
    cursor cCanc is
    select a.numero_guia, b.clvsituacion, b.rowid reng_guia, c.rowid reng_paquete
    from uo_cancelaciones a, guia@cdb b, paquete@cdb c
    where a.numero_guia = b.numero_guia
    and a.numero_guia = c.numero_guia
    and a.cancelado is null
    and a.cancelacion_denegada is null;

    verror_cod          varchar2(15);
    verror_msg          varchar2(80);

begin
    for vCanc in cCanc
    loop
        begin
            If vCanc.clvsituacion not in (6,7,1,3,20,109) then
                update guia@cdb set
                    clvsituacion = 20,
                    fechacancelacion = sysdate,
                    estaactiva = 0
                    where rowid = vCanc.reng_guia;
                update paquete@cdb set
                    clvsituacion = 20
                    where rowid = vCanc.reng_paquete;
                update uo_cancelaciones set
                    fechacancelado = sysdate,
                    cancelado = 'X'
                    where numero_guia = vCanc.numero_guia;

                commit;

            else
                update uo_cancelaciones set
                    cancelacion_denegada = 'X'
                    where numero_guia = vCanc.numero_guia;

                commit;

            end if;
        exception
        when others then
            begin
                verror_cod := to_char(sqlcode);
                verror_msg := substr(sqlerrm,1,100);
                insert into uo_cancelaciones_probls (
                    error,
                    fecha_error,
                    cancelacion_solicitada)
                    values (substr(verror_cod||': '||verror_msg,1,100),
                        sysdate,
                        'X');

                commit;

            end;
        end;
    end loop;
declare
    cursor cVenc is
    select a.numero_guia,
    to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr') dias_vigencia
    from uo_regreso a, guia@cdb b
    where a.numero_guia = b.numero_guia
    and to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr') >= 16
    and b.clvsituacion not in (1, 7, 51, 57, 63, 20)
    and a.vencido is null
    group by a.numero_guia, to_date(sysdate,'dd/mm/rr') - to_date(a.fechadocumentacion,'dd/mm/rr');
```

```

for vVenc in cVenc
loop
    begin
        update guia@cdb set
            clvsituacion = 20,
            fechacancelacion = sysdate,
            estaactiva = 0,
            observaciones = 'ENVIO VENCIDO'
        where numeroguia = vVenc.numeroguia;
        update paquete@cdb set
            clvsituacion = 20
        where numeroguia = vVenc.numeroguia;
        commit;
    exception
    when others then
        begin
            verror_cod := to_char(sqlcode);
            verror_msg := substr(sqlerrm,1,100);
            insert into uo_cancelaciones_probls (
                error,
                fecha_error,
                cancelacion_vencimiento)
            values (substr(veerror_cod||': '|verror_msg,1,100),
                sysdate,
                'X');
        end;
    end;
end loop;
end;
exception
when others then null;
end;
/
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') from dual;
spool off
exit

```

## E. Proceso para Informar situaciones.

```

spool c:\uol\logs\uo_regreso.log
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') from dual;
declare
    cursor c_situacion is
    select
        a.referenciacliente
        , a.numeroguia
        , a.clvsituacion
        , a.fechasituacion
        , b.clvcausanoentrega
        , a.fechadocumentacionorigen
        , a.fechaentrega
        , a.nombrepersonarecibe
        , substr(a.folioidentificacion,1,20) folioidentificacion
        , substr(c.descripcion,1,30) descripcion
        , decode(substr(upper(a.nombrepersonarecibe),1,10),null,null,substr(upper(d.nombre),1,10),1,2) consignatariooriginal
        , a.observaciones
    from guia@prod a
        , paquete@prod b
        , tipoidentificacion@prod c
        , consignatario@prod d
    where
        a.numeroguia like '7309%'
        and a.numeroguia = b.numeroguia
        and a.clvtipoidentificacion = c.clvtipoidentificacion(+)
        and a.clvconsignatario = d.clvconsignatario
        and to_date(a.fechasituacion,'dd/mm/rr') > to_date(sysdate,'dd/mm/rr')-2;

    v_primer number;
    v_ultimo number;
    v_guardar number := 1;
    v_contar number := 1;
begin
    for v_situacion in c_situacion
    loop
        begin
            insert into uo_regreso(
                guia_uo,
                numeroguia,
                causanoentrega,
                situacion,
                fechasituacion,
                fechaentrega,
                fechain1,
                fechain2,
                nombrecibe,
                folioidentrecibe,
                identrecibe,
                consignatariooriginal,
                fechadocumentacion,
                vencido)
            values(
                v_situacion.referenciacliente,
                v_situacion.numeroguia,
                v_situacion.clvcausanoentrega,
                v_situacion.clvsituacion,
                v_situacion.fechasituacion,
                v_situacion.fechaentrega,

                decode(v_situacion.clvsituacion,102,v_situacion.fechasituacion,8,v_situacion.fechasituacion,null),
                decode(v_situacion.clvsituacion,103,v_situacion.fechasituacion,null),
                v_situacion.nombrepersonarecibe,
                v_situacion.folioidentificacion,
                v_situacion.descripcion,
                v_situacion.consignatariooriginal,

```

```

v_situacion.fechadocumentacionorigen,
decode(v_situacion.observaciones,'ENVIO VENCIDO','X',null)
);
exception
when no_data_found then
exit;
when others then null;
end;
if
v_contar > v_guardar then
v_contar := 1;
commit;
else
v_contar := v_contar + 1;
end if;
end loop;
commit;

declare
vArchivo          utl_file.file_type;
vArchftp          utl_file.file_type;
vBuffer           varchar2(500);
vCuenta          number(5) := 0;
vTotEntEfe       number(5) := 0;
vImpEntEfe       number(5) := 0;
vTotEntChq       number(5) := 0;
vImpEntChq       number(5) := 0;
cursor c_genera is
select
a.rowid renglon,
decode(a.situacion,
1,100,
7,100,
51,100,
57,100,
63,100,
13,decode(a.vencido,'X',106,102),
20,decode(a.vencido,'X',106,102),
35,decode(a.vencido,'X',106,102),
53,decode(a.vencido,'X',106,102),
62,decode(a.vencido,'X',106,102),
101,decode(a.vencido,'X',106,102),
104,decode(a.vencido,'X',106,102),
106,decode(a.vencido,'X',106,102),
107,decode(a.vencido,'X',106,102),
5,103,
34,103,
58,103,
8,decode(a.causanoentrega,7,103,5,103,104),
102,104,
103,105,
10,106,
37,106,
108,106,
101) situacion,
a.guia_uo,
b.numero_ident,
b.transtype,
b.datetime,
b.amount_pesos,
b.tipo_moneda,
a.fechaint1,
a.fechaint2,
a.fechasituacion,
a.fechaentrega,
b.nombre_bvro,
b.nombre2_bvro,
b.direccion,
b.colonia,
b.ciudad,
b.estado,

```

```

b.prefijo,
b.telefono,
b.codigo_postal,
b.remitente,
a.folioidentrecibe,
a.identrecibe,
a.consignatariooriginal,
a.transmitido,
a.causanoentrega,
a.vencido
from uo_regreso a, uo_transf b
where a.guia_uo = b.numero_guia
and a.transmitido is null
order by situacion, b.transtype, a.guia_uo;

begin
vArchivo := utl_file.fopen('c:\uo\Regresouo', 'w', 'to_char(sysdate, 'mmdd') || '01.txt', 'w');
for v_genera in c_genera
loop
declare
cursor c_ultima_sit is
select
        guia_uo,
        max(fechasituacion) maxfechasit
from uo_regreso
where guia_uo = v_genera.guia_uo
group by guia_uo;
v_guia          varchar2(15);
v_fecha         date;

begin
open c_ultima_sit;
fetch c_ultima_sit into v_guia, v_fecha;
close c_ultima_sit;
if v_genera.guia_uo = v_guia and v_genera.fechasituacion = v_fecha and v_genera.transmitido is
null then
vBuffer := rpad(''||to_char(v_genera.situacion)
||', '||v_genera.guia_uo
||', '||v_genera.numero_ident
||', '||v_genera.transtype
||', '||to_char(v_genera.datetime, 'dd-mm-yyyy hh24:mi')
||', '||to_char(v_genera.amount_pesos)
||'.00, '||v_genera.tipo_moneda
||', '||to_char(v_genera.fechaint1, 'dd-mm-yyyy hh24:mi')
||', '||to_char(v_genera.fechaint2, 'dd-mm-yyyy hh24:mi')
||', '||to_char(v_genera.fechaentrega, 'dd-mm-yyyy hh24:mi')
||', '||v_genera.nombre_buro
||', '||v_genera.direccion
||', '||v_genera.colonia
||', '||v_genera.ciudad
||', '||v_genera.estado
||', '||v_genera.prefijo
||', '||v_genera.telefono
||', '||v_genera.codigo_postal
||', '||v_genera.remitente
||', '||v_genera.folioidentrecibe
||', '||v_genera.identrecibe
||', '||to_char(v_genera.consignatariooriginal)
||', '500, ');
utl_file.put_line(vArchivo, vBuffer);
if v_genera.situacion = 100 then
if v_genera.transtype = 'EFE' then
vTotEntEfe := vTotEntEfe + 1;
vImpEntEfe := vImpEntEfe + v_genera.amount_pesos;
else
vTotEntChq := vTotEntChq + 1;
vImpEntChq := vImpEntChq + v_genera.amount_pesos;
end if;
end if;
if v_genera.situacion in (100, 102, 106) then
update uo_regreso set transmitido = 'X' where guia_uo = v_genera.guia_uo;

```



## REFERENCIAS ELECTRONICAS.

[http://www.ciscosystems.co.nz/warp/public/cc/so/cuso/sp/teico/dcn/sp\\_26kdc\\_ds.htm](http://www.ciscosystems.co.nz/warp/public/cc/so/cuso/sp/teico/dcn/sp_26kdc_ds.htm)

<http://www.telecos-malaga.com/apuntes.php?asignatura=redeso>

<http://personales.mundivia.es/astruc/doctxt30.htm>

<http://www.inaoep.mx/~moises/AGC/sw-rout.html>

<http://www.metalink.oracle.com>

<http://www2.westernunion.com>

<http://www.multipack.com.mx>

## BIBLIOGRAFIA.

### ORACLE8 .Programación PL/SQL

Scott Urman

McGraw Hill/Interamericana de España S. A. U.

Edificio Valrealty, 1º planta

Basaurí, 17

28023 Aravaca (Madrid).

### ORACLE8. Manual del Administrador.

Kevin Loney

Osborne-McGraw Hill

Oficinas Iberoamericanas

McGraw Hill/Interamericana Editores, S.A. de C.V.

Atlacomulco 499-501

Fracc. Ind. Sn. Andrés Atoto 53500

Naucalpan de Juarez Edo de México.

### El Gran libro de Windows NT 4.

Bernd Kretschmer

Marcambo Boixarev Editores

Barcelona, 1996.